

Springer
Handbook *of*
**Engineering
Statistics**

Pham
Editor

2nd Edition

Springer Handbooks

Springer Handbooks maintain the highest standards of references in key areas of the physical and applied sciences for practitioners in industry and academia, as well as graduate students. Designed to be useful and readable desk reference books, but also prepared in various electronic formats, these titles allow fast yet comprehensive review and easy retrieval of essential reliable key information. Springer Handbooks cover methods, general principles, functional relationships and fundamental data and review established applications.

All Springer Handbooks are edited and prepared with great care by editors committed to harmonizing the content. All chapters are written by international experts in their field.

Indexed by SCOPUS. The books of the series are submitted for indexing to Web of Science.

Hoang Pham
Editor

Springer Handbook of Engineering Statistics

Second Edition

With 400 Figures and 203 Tables

 Springer

Editor
Hoang Pham
Department of Industrial & Systems Engineering
Rutgers University
Piscataway, NJ, USA

ISSN 2522-8692 ISSN 2522-8706 (electronic)
Springer Handbooks
ISBN 978-1-4471-7502-5 ISBN 978-1-4471-7503-2 (eBook)
<https://doi.org/10.1007/978-1-4471-7503-2>

1st edition: © Springer-Verlag London Limited 2006

© Springer-Verlag London Ltd., part of Springer Nature 2023, corrected publication 2023

The author(s) has/have asserted their right(s) to be identified as the author(s) of this work in accordance with the Copyright, Designs and Patents Act 1988.

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer-Verlag London Ltd. part of Springer Nature. The registered company address is: The Campus, 4 Crinan Street, London, N1 9XW, United Kingdom

To My Mother on Her 90th Birthday!

Preface

In recent years modern techniques of statistics have been applied to a wide variety of fields and applications especially in big data in health informatics, machine learning, and signal processing. Statistical techniques are finding increased use in the fields of engineering, data science, artificial intelligence (AI), and Industry 4.0 technologies and will continue even more so in the coming decades.

In response to such technological trends along with much feedback that we have received from users of the first edition of the *Springer Handbook of Engineering Statistics* (published in 2006), we have made substantial changes in this new edition by

- Adding 35 completely new contributions revolving around current trends related to modern statistical computing and its applications ranging from data science to electronic packaging, from high-dimensional data to AI, and from fusion data to optimal decision-making
- Updating more than 20 selected chapters from the first edition thoroughly to reflect the current state of the art where all the topics from the previous edition have integrated into all the chapters in this new edition

The *Springer Handbook of Engineering Statistics*, altogether 56 chapters, provides a comprehensive state-of-the-art reference volume that covers both fundamental and theoretical work in the areas of engineering statistics including: adaptive progressive censoring, AI, accelerated life testing, adversarial learning, artificial neural network, Bayesian inferences and prediction, Bootstrap and boosting methods, censored survival data, competing risk models, change-point detection, correlated failures, counting processes, copula methods, data mining, deep learning, dynamic robust design, difference of convex functions (DC), detecting outliers in high-dimensional data, dimension reduction and nonparametric methods, distributional inference, electronic packaging, fatigue life prediction, functional dependence measure, fusion learning, genetic algorithms, exponentially weighted moving average, gradient method, high-dimension time series, importance measures, indecisive weighted voting system, logistic regression, machine learning, multi-agent support systems, majorization-minimization, Monte Carlo simulation, influential observations, Markov decision process, model selection, multivariate distributions, multiple stresses designs, multivariate time series, multi-criteria decisions, nonparametric empirical Bayesian, inspection-maintenance, order statistics, product reliability, repairable system reliability, renewal processes, process control and improvement, proportional hazards regression and sampling, random forests for censoring data, recursive partitioning algorithm, software reliability, semi-parametric for survival data, step-stress life testing, statistical process control, six sigma design and methodology, stationary causal process, stochastic dependences, statistical design and diagnostics, stochastic expectation maximization algorithm, stress-strength models, unsupervised learning, variable importance in high-dimensional, voltage, and vibration acceleration models, among others.

The contents of the handbook are organized in six parts:

- PART I. Fundamental Statistics and Its Applications
- PART II. Process Monitoring and Improvement
- PART III. Reliability Models and Survival Analysis
- PART IV. Advanced Statistical Methods and Modeling
- PART V. Statistical Computing and Data Mining
- PART VI. Applications in Engineering Statistics

The chapters are written by more than 110 leading experts in statistics, biostatistics, engineering statistics, reliability engineering, computer science, applied statistics, machine learning, management, and related areas.

Practitioners will certainly find this *Springer Handbook* useful when looking for solutions to practical problems. Researchers, statisticians, scientists, engineers, teachers, and students can use it for quick access to the background, recent research and trends, and most important references regarding certain topics, if not all, in engineering statistics.

Acknowledgments I am deeply indebted to our authors from 20 different countries for the hard work, time, and effort that they put into their contributions, especially during the COVID-19 pandemic.

My grateful appreciation goes to the international advisory board members who have provided excellent help for this book. I would also like to express my thanks to the editorial board members who have reviewed and offered a variety of constructive criticisms on the chapters in the handbook.

I wish to thank our editors from Springer: Dr. Judith Hinterberg, Anthony Doyle, and Ursula Barth for their support and guidance. It has been a pleasure working with them, and I thank Springer for their commitment to publish this new edition.

I am indebted to my wife and children for permanent support and patience.

Piscataway, New Jersey
July 2022

Hoang Pham

Contents

Part I Fundamental Statistics and Its Applications	1
1 Basic Statistics	3
Hoang Pham	
2 Generalized Statistical Distributions	43
Gauss M. Cordeiro and Artur Lemonte	
3 Statistics for Reliability Modeling	53
Paul Kvam and Jye-Chyi Lu	
4 Functional Data Analysis	67
Yuhang Xu	
5 Symmetric Geometric Skew Normal Regression Model	87
Debasis Kundu and Deepak Prajapati	
6 Statistical Analysis of Modern Reliability Data	105
Yueyao Wang, I-Chen Lee, Lu Lu, and Yili Hong	
7 Mathematical Reliability Aspects of Multivariate Probability Distributions	129
Lidia Z. Filus and Jerzy K. Filus	
8 Introduction to Stochastic Processes	137
Hoang Pham	
9 Progressive Censoring Methodology	153
Narayanaswamy Balakrishnan and Erhard Cramer	
10 Warranty Policies: Analysis and Perspectives	185
Hoang Pham and Jun Bai	
Part II Process Monitoring and Improvement	197
11 Statistical Methods for Quality and Productivity Improvement	199
Wei Jiang, Terrence E. Murphy, Kwok-Leung Tsui, and Yang Zhao	
12 Chain Sampling	221
Govindaraju Kondaswamy	
13 Six Sigma	239
Fugee Tsung and Kai Wang	
14 Statistical Models for Monitoring the High-Quality Processes	261
Min Xie, Thong Ngee Goh, and Tahir Mahmood	
15 Statistical Management and Modeling for Demand Spare Parts	275
Emilio Ferrari, Arrigo Pareschi, Alberto Regattieri, and Alessandro Persona	

16	D-Efficient Mixed-Level Foldover Designs for Screening Experiments	305
	Nam-Ky Nguyen, Ron S. Kenett, Tung-Dinh Pham, and Mai Phuong Vuong	
17	Censored Data Prediction Based on Model Selection Approaches	315
	Tzong-Ru Tsai, Jyun-You Chiang, Shuai Wang, and Yan Qin	
18	Monitoring Coefficient of Variation Using CUSUM Control Charts	333
	Phuong Hanh Tran, Huu Du Nguyen, Cédric Heuchenne, and Kim Phuc Tran	
19	Change-Point-Based Statistical Process Controls	361
	Partha Sarathi Mukherjee	
Part III Reliability Models and Survival Analysis		383
20	Reliability Characteristics and Optimization of Warm Standby Systems	385
	Suprasad V. Amari and Hoang Pham	
21	Importance and Sensitivity Analysis on Complex Repairable Systems and Imprecise System Reliability	403
	Geng Feng	
22	Hardware and Software Reliability, Verification, and Testing	415
	Ashis Kumar Chakraborty, E. V. Gijo, Anisha Das, and Moutushi Chatterjee	
23	OSS Reliability Analysis and Project Effort Estimation Based on Computational Intelligence	443
	Shigeru Yamada, Yoshinobu Tamura, and Kodai Sugisaki	
24	Vulnerability Discovery Analysis in Software Reliability and Related Optimization Problems	457
	P. K. Kapur and Saurabh Panwar	
25	Software Reliability Modeling and Prediction	481
	Hoang Pham and Xiaolin Teng	
26	Statistical Maintenance Modeling	495
	Hoang Pham and Wenjian Li	
27	Continuous-Time Predictive Maintenance Modeling with Dynamic Decision Framework	521
	Antoine Grall and Elham Mosayebi Omshi	
28	Stochastic Redundant Replacement Maintenance Models	543
	Toshio Nakagawa, Satoshi Mizutani, and Xufeng Zhao	
Part IV Advanced Statistical Methods and Modeling		573
29	Confidence Distribution and Distribution Estimation for Modern Statistical Inference	575
	Yifan Cui and Min-ge Xie	
30	Logistic Regression Tree Analysis	593
	Wei-Yin Loh	
31	Detecting Outliers and Influential and Sensitive Observations in Linear Regression	605
	Daniel Peña	
32	Statistical Methodologies for Analyzing Genomic Data	621
	Fenghai Duan and Heping Zhang	

33 Genetic Algorithms and Their Applications	635
Mitsuo Gen and Lin Lin	
34 Deterministic and Stochastic DCA for DC Programming	675
Hoai An Le Thi, Tao Pham Dinh, Hoang Phuc Hau Luu, and Hoai Minh Le	
35 Inference for Coherent Systems with Weibull Components Under a Simple Step-Stress Model	703
Narayanaswamy Balakrishnan, Debanjan Mitra, and Xiaojun Zhu	
36 Bivariate Distributions with Singular Components	733
Debasis Kundu	
37 Bayesian Models	763
Ashis Kumar Chakraborty, Soumen Dey, Poulami Chakraborty, and Aleena Chanda	
Part V Statistical Computing and Data Mining	795
38 Data Mining Methods and Applications	797
Kwok-Leung Tsui, Victoria Chen, Wei Jiang, Fangfang Yang, and Chen Kan	
39 Statistical Methods for Tensor Data Analysis	817
Qing Mai and Xin Zhang	
40 Random Forests for Survival Analysis and High-Dimensional Data	831
Sarah E. Formentini, Yifan Cui, and Ruoqing Zhu	
41 Probability Inequalities for High-Dimensional Time Series Under a Triangular Array Framework	849
Fang Han and Wei Biao Wu	
42 Statistical Machine Learning	865
Maryam Arabzadeh Jamali and Hoang Pham	
43 Covariance Estimation via the Modified Cholesky Decomposition	887
Xiaoning Kang, Zhiyang Zhang, and Xinwei Deng	
44 Statistical Learning	901
Qingyi Gao and Xiao Wang	
45 Bayesian Survival Analysis in the Presence of Monotone Likelihoods	921
Jing Wu, Mário de Castro, and Ming-Hui Chen	
46 Multivariate Modeling with Copulas and Engineering Applications	931
Jun Yan	
Part VI Applications in Engineering Statistics	947
47 Environmental Risks Analysis Using Satellite Data	949
Yuriy V. Kostyuchenko	
48 Probabilistic Models for Reliability Analysis Using Safe-Life and Damage Tolerance Methods	965
Xuefei Guan and Jingjing He	
49 Application of Cognitive Architecture in Multi-Agent Financial Decision Support Systems	981
Marcin Hernes and Ngoc Thanh Nguyen	
50 Reliability of Electronic Packaging	1003
Wen-Fang Wu and Yi-An Lu	

51 Accelerated Life Testing	1025
Qingchuan He, Wen-Hua Chen, and Jun Pan	
52 Accelerated Life Testing Data Analyses for One-Shot Devices	1039
Narayanaswamy Balakrishnan and Man Ho Ling	
53 Tangent Space Approximation in Geometric Statistics	1059
Ted Chang	
54 Statistical Modeling of Discrete Choices for Human Behaviors	1075
Xi Zhu and Shuai Huang	
55 Weighted Voting Systems	1089
Hainan Zhang	
56 Image Registration, Rigid Bodies, and Unknown Coordinate Systems	1109
Ted Chang	
Correction to: Monitoring Coefficient of Variation Using CUSUM Control Charts	C1
Phuong Hanh Tran, Huu Du Nguyen, Cédric Heuchenne, and Kim Phuc Tran	
Index	1129

About the Editor



Hoang Pham is a distinguished professor and former chairman (2007–2013) of the Department of Industrial and Systems Engineering at Rutgers University. Before joining Rutgers in 1993, he was a senior engineering specialist with the Idaho National Engineering Laboratory, Idaho Falls, Idaho, and Boeing Company in Seattle, Washington. His research areas include reliability modeling and prediction, software reliability, and statistical inference. He is editor-in-chief of the *International Journal of Reliability, Quality and Safety Engineering* and editor of Springer Series in Reliability Engineering and has been conference chair and program chair of over 40 international conferences and workshops. Dr. Pham is the author or coauthor of 7 books and has published over 200 journal articles, 100 conference papers, and edited 20 books including *Springer Handbook of Engineering Statistics* and *Handbook of Reliability Engineering*. He has delivered over 40 invited keynote and plenary speeches at many international conferences and institutions. His numerous awards include the 2009 IEEE Reliability Society Engineer of the Year Award. He is a fellow of the IEEE, AAIA, and IISE.

International Advisory Board

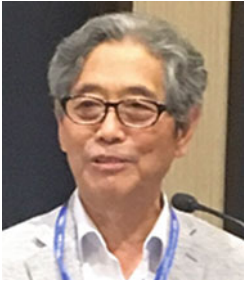


Narayanaswamy Balakrishnan

Department of Mathematics and Statistics
McMaster University
Hamilton, ON, Canada

Dr. Narayanaswamy Balakrishnan is a distinguished university professor in the Department of Mathematics and Statistics at McMaster University, Hamilton, Ontario, Canada. He received his B.Sc. and M.Sc. in Statistics from the University of Madras, India, in 1976 and 1978, respectively, and then his Ph.D. from the Indian Institute of Technology, Kanpur, India, in 1981. He has very wide research interests spanning univariate and multivariate distribution theory, reliability analysis, ordered data analysis, biostatistics and survival analysis, statistical inference, applied probability, nonparametric statistics, and statistical methods and applications to different fields in science and engineering, and has published many journal articles in all these areas. He has coauthored numerous books including the well-known four volumes on distribution theory published by John Wiley & Sons and has also coedited the Second Edition of 16-Volume *Encyclopedia of Statistical Sciences* published by John Wiley & Sons. He is the editor-in-chief of *Communications in Statistics* and *Mathematical Methods of Statistics* and is an associate editor of numerous journals such as *Journal of Multivariate Analysis*, *Statistical Methods in Medical Research*, *International Statistical Review*, *Probability in the Engineering and Informational Sciences*, and *Applied Stochastic Models in Business and Industry*.

He has received many honors and accolades including fellow of the Royal Society of Canada, fellow of the American Statistical Association, and fellow of the Institute of Mathematical Statistics and an honorary doctorate degree from the National and Kapodistrian University of Athens, Greece.

**Toshio Nakagawa**

Department of Business Administration
Aichi Institute of Technology
Toyota, Japan

Dr. Toshio Nakagawa received B.S.E. and M.S. degrees from Nagoya Institute of Technology in 1965 and 1967, respectively, and Doctor of Engineering degree from Kyoto University in 1977. He worked as a research associate at Syracuse University for 2 years from 1972 to 1973. He is now a honorary professor at Aichi Institute of Technology in Toyota City. He has published 7 books with Springer, 20 book chapters, and about 200 journal papers. His research interests are optimization problems in operations research and management science and analysis for stochastic and computer systems in reliability and maintenance theory.

**Min-ge Xie**

Rutgers University
Piscataway, NJ, USA

Dr. Min-ge Xie is a distinguished professor and director of the Office of Statistical Consulting, Department of Statistics, Rutgers University, the State University of New Jersey. His main research interest is in the foundation of statistical inference, fusion learning, and data science. His other expertise includes estimating equations, robust statistics, hierarchical models, asymptotic theories, and applications in biostatistics and industry. Dr. Xie received his B.S. degree in mathematics from the University of Science and Technology (USTC) with the highest honor and Ph.D. degree in statistics from the University of Illinois at Urbana-Champaign (UIUC). He is a fellow of the American Statistical Association, a fellow of the Institute of Mathematical Statistics, and an elected member of the International Statistical Institute. He has served on numerous scientific review panels and editorial boards. He is a co-founder of the BFF research community. His research has been supported in part by grants from NSF, NIH, DHS, NSA, and FAA.

International Editorial Board



Mingxiao Jiang

Medtronic, PLC.
Minneapolis, MN, USA

Dr. Mingxiao Jiang received his B.S. in Engineering Mechanics from Zhejiang University, M.S. in Reliability Engineering from the University of Arizona, and Ph.D. in Mechanics of Materials from Georgia Institute of Technology. He is currently a technical fellow and a distinguished engineer at Medtronic. He is a CRE and a senior member of ASQ and IEEE. He has 3 US patents and 50 publications in refereed journals and conference proceedings. He has been active in reliability societies as a vice chair for the organization RAMS® (2009–2013) and committee member for several international reliability conferences. He has been serving on the Editorial Review Boards for Reliability Review and Quality Engineering.



Yi-Kuei Lin

National Yang Ming Chiao Tung University
Hsinchu, Taiwan

Dr. Yi-Kuei Lin is currently a chair professor of the Industrial Engineering and Management Department, National Yang Ming Chiao Tung University, Taiwan. He was the president of the Operations Research Society of Taiwan in 2018 and 2019. He received a bachelor degree from the Applied Mathematics Department at National Chiao Tung University, Taiwan. He obtained his master degree and Ph.D. degree from the Department of Industrial Engineering and Engineering Management at National Tsing Hua University, Taiwan, Republic of China. He had the honor of receiving the Outstanding Research Awards from the Ministry of Science and Technology of Taiwan in 2008, 2010, and 2013, respectively. He served as dean, College of Management at Vanung University from 2003 to 2007, and chairman, Department of Industrial Management at National Taiwan University of Science and Technology from 2012 to 2014. He is now serving as associate editor of *IEEE Transactions on Reliability*, *Annals of Operations Research*, and *Quality Technology & Quantitative Management*. His research interest includes performance evaluation, network reliability, operations research, and telecommunication management. He has published more than 200 papers in refereed international journals



Hon Keung Tony Ng
Southern Methodist University
Dallas, TX, USA

Dr. Hon Keung Tony Ng received his Ph.D. in Mathematics from McMaster University, Hamilton, ON, Canada, in 2002. He is currently a professor with the Department of Statistical Science, Southern Methodist University, Dallas, TX, USA. His research interests include reliability, censoring methodology, ordered data analysis, nonparametric methods, and statistical inference. Dr. Ng is an associate editor of *Communications in Statistics*, *Computational Statistics*, *IEEE Transactions on Reliability*, *Journal of Statistical Computation and Simulation*, *Naval Research Logistics*, *Sequential Analysis*, and *Statistics & Probability Letters*. He is the coeditor of the books *Ordered Data Analysis*, *Modeling and Health Research Methods*, *Statistical Modeling for Degradation Data*, *Statistical Quality Technologies: Theory and Practice*, and *Advances in Statistics: Theory and Applications* published by Springer. He is a fellow of the American Statistical Association and an elected member of the International Statistical Institute.



Peihua Qiu
University of Florida
Gainesville, FL, USA

Dr. Peihua Qiu received his Ph.D. in Statistics from the Department of Statistics at the University of Wisconsin at Madison in 1996. He worked as a senior research consulting statistician of the Biostatistics Center at the Ohio State University during 1996–1998. He then worked as an assistant professor (1998–2002), an associate professor (2002–2007), and a full professor (2007–2013) at the School of Statistics at the University of Minnesota. He is an elected fellow of the American Statistical Association, an elected fellow of the Institute of Mathematical Statistics, and an elected member of the International Statistical Institute. He served as associate editor for some top statistical journals, including *Journal of the American Statistical Association*, *Biometrics*, and *Technometrics*. He was the editor (2014–2016) of *Technometrics*. Since 2013, he has been a professor and the founding chair of the Department of Biostatistics at the University of Florida. Qiu has made substantial contributions in the areas of jump regression analysis, image processing, statistical process control, survival analysis, disease screening, and disease surveillance. So far, he has published over 140 research papers in refereed journals, many of which appeared in top journals, such as *Technometrics*, *Journal of the American Statistical Association*, and *IEEE Transactions on Pattern Analysis and Machine Intelligence*. His research monograph titled *Image Processing and Jump Regression Analysis* (2005, Wiley) won the inaugural Ziegel prize in 2007. His second book titled *Introduction to Statistical Process Control* was published in 2014 by Chapman & Hall/CRC.

**Mengmeng Zhu**

North Carolina State University
Raleigh, NC, USA

Dr. Mengmeng Zhu is an assistant professor in the Department of Textile Engineering, Chemistry and Science, and an associate faculty member of the Operations Research Graduate Program, North Carolina State University, Raleigh, North Carolina, USA. She received the Ph.D. degree in Industrial and Systems Engineering and the M.S. degrees in Statistics and Industrial and Systems Engineering from Rutgers University, New Brunswick, New Jersey, USA, in 2018, 2016, and 2015, respectively. Her research interests lie at system reliability and resilience engineering, applied operations research, prognostics and health management, and computational data analytics. Dr. Zhu is the recipient of the Best Paper Award of the 24th ISSAT International Conference on Reliability and Quality in Design, 2018, and the Best Student Paper Award of the 23rd ISSAT International Conference on Reliability and Quality in Design, 2017.

Contributors

Suprasad V. Amari BAE Systems, Merrimack, NH, USA

Maryam Arabzadeh Jamali Department of Industrial and Systems Engineering, Rutgers University, Piscataway, NJ, USA

Jun Bai Barclays Bank US, Wilmington, DE, USA

Narayanaswamy Balakrishnan Department of Mathematics and Statistics, McMaster University, Hamilton, ON, Canada

Mário de Castro Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, Brazil

Ashis Kumar Chakraborty Indian Statistical Institute, Kolkata, West Bengal, India

Poulami Chakraborty Indian Statistical Institute, Kolkata, West Bengal, India

Aleena Chanda Indian Statistical Institute, Kolkata, West Bengal, India

Ted Chang Department of Statistics, University of Virginia, Charlottesville, VA, USA

Moutushi Chatterjee Lady Brabourne College, Kolkata, West Bengal, India

Ming-Hui Chen Department of Statistics, University of Connecticut, Storrs, CT, USA

Victoria Chen Department of Industrial, Manufacturing, & Systems Engineering, University of Texas at Arlington, Arlington, TX, USA

Wen-Hua Chen School of Mechanical Engineering, Zhejiang Sci-Tech University, Hangzhou, China

Jyun-You Chiang School of Statistics, Southwestern University of Finance and Economics, Chengdu, China

Gauss M. Cordeiro UFPE, Recife, Brazil

Erhard Cramer Institute of Statistics, RWTH Aachen University, Aachen, Germany

Yifan Cui Zhejiang University, Hangzhou, Zhejiang, China

Anisha Das Indian Statistical Institute, Kolkata, West Bengal, India

Xinwei Deng Department of Statistics, Virginia Tech, Blacksburg, VA, USA

Soumen Dey Indian Statistical Institute, Kolkata, West Bengal, India

Tao Pham Dinh National Institute of Applied Sciences – Rouen, Rouen, France

Fenghai Duan Department of Biostatistics, Brown University School of Public Health, Providence, RI, USA

Geng Feng School of Engineering, University of Central Lancashire, Preston, UK

Emilio Ferrari Department of Industrial Engineering, University of Bologna, Bologna, Italy

Jerzy K. Filus Department of Mathematics and Computer Science, Oakton Community College, Des Plaines, IL, USA

Lidia Z. Filus Department of Mathematics, Northeastern Illinois University, Chicago, IL, USA

Sarah E. Formentini University of Illinois at Urbana-Champaign, Champaign, IL, USA

Qingyi Gao Purdue University, West Lafayette, IN, USA

Mitsuo Gen Fuzzy Logic Systems Institute (FLSI), Department of Research, Fukuoka, Japan
Tokyo University of Science (TUS), Research Institute of Science & Technology, Tokyo, Japan

E. V. Gijo Indian Statistical Institute, Bangalore, Karnataka, India

Thong Ngee Goh Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore, Republic of Singapore

Antoine Grall LIST3N, Université de Technologie de Troyes, Champagne, de Troyes, France

Xuefei Guan Graduate School of China Academy of Engineering Physics, Beijing, China

Fang Han University of Washington, Seattle, WA, USA

Jingjing He School of Reliability and Systems Engineering, Beihang University, Beijing, China

Qingchuan He School of Mechanical Engineering, Zhejiang Sci-Tech University, Hangzhou, China

Marcin Hernes Faculty of Management, Wroclaw University of Economics and Business, Wroclaw, Poland

Cédric Heuchenne HEC-Liège, Management School of the University of Liège, Liège, Belgium

Yili Hong Department of Statistics, Virginia Tech, Blacksburg, VA, USA

Shuai Huang University of Washington, Seattle, WA, USA

Wei Jiang Antai College of Economics and Management, Shanghai Jiao Tong University, Shanghai, China

Chen Kan Department of Industrial, Manufacturing, & Systems Engineering, University of Texas at Arlington, Arlington, TX, USA

Xiaoning Kang International Business College and Institute of Supply Chain Analytics, Dongbei University of Finance and Economics, Dalian, China

P. K. Kapur Amity Center for Interdisciplinary Research, Amity University, Noida, Uttar Pradesh, India

Ron S. Kenett KPA Ltd., Samuel Neaman Institute, Technion, Israel

Govindaraju Kondaswamy School of Mathematical and Computational Sciences, Massey University, Palmerston North, New Zealand

Yuriy V. Kostyuchenko Systemic Risk and Resilience Research Group, Advancing Systems Analysis Program, International Institute for Applied Systems Analysis (IIASA), Laxenburg, Austria

Debasis Kundu Department of Mathematics and Statistics, Indian Institute of Technology Kanpur, Kanpur, Uttar Pradesh, India

Paul Kvam Department of Mathematics and Computer Science, University of Richmond, Richmond, VA, USA

I-Chen Lee Department of Statistics, National Cheng Kung University, Tainan, Taiwan

Artur Lemonte UFRN, Natal, Brazil

Hoai Minh Le University of Lorraine, Metz, France

Institut Universitaire de France (IUF), Paris, France

Wenjian Li Marketing Science, Javelin Direct, Inc., Irving, TX, USA

Lin Lin Dalian University of Technology, Economy and Technology Development Area, Dalian, China

Man Ho Ling The Education University of Hong Kong, Hong Kong, China

Wei-Yin Loh Department of Statistics, University of Wisconsin, Madison, WI, USA

Lu Lu Department of Mathematics and Statistics, University of South Florida, Tampa, FL, USA

Jye-Chyi Lu The School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA

Yi-An Lu Department of Mechanical Engineering, National Taiwan University, Taipei, Taiwan

Hoang Phuc Hau Luu University of Lorraine, Metz, France

Institut Universitaire de France (IUF), Paris, France

Tahir Mahmood Industrial and Systems Engineering Department, College of Computing and Mathematics, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

Qing Mai Department of Statistics, Florida State University, Tallahassee, FL, USA

Debanjan Mitra Indian Institute of Management Udaipur, Udaipur, Rajasthan, India

Satoshi Mizutani Department of Business Administration, Aichi Institute of Technology, Aichi, Japan

Elham Mosayebi Omshi LIST3N, Université de Technologie de Troyes, Champagne, de Troyes, France

Partha Sarathi Mukherjee Interdisciplinary Statistical Research Unit (ISRU), Indian Statistical Institute (ISI), Kolkata, West Bengal, India

Terrence E. Murphy Biostatistics and Bioinformatics Department of Public Health Sciences, Pennsylvania State University College of Medicine, Hershey, PA, USA

Toshio Nakagawa Department of Business Administration, Aichi Institute of Technology, Aichi, Japan

Huu Du Nguyen International Research Institute for Artificial Intelligence and Data Science, Dong A University, Danang, Vietnam

Nam-Ky Nguyen Vietnam Institute for Advanced Study in Mathematics, Hanoi, Vietnam

Ngoc Thanh Nguyen Faculty of Information and Communication Technology, University of Science and Technology, Wrocław, Poland

Jun Pan School of Mechanical Engineering, Zhejiang Sci-Tech University, Hangzhou, China

- Saurabh Panwar** Department of Operational Research, University of Delhi, Delhi, India
- Arrigo Pareschi** Department of Industrial Engineering, University of Bologna, Bologna, Italy
- Daniel Peña** Department of Statistics, Universidad Carlos III de Madrid, Madrid, Spain
- Alessandro Persona** Department of Management and Engineering, University of Padova, Vicenza, Italy
- Hoang Pham** Department of Industrial and Systems Engineering, Rutgers University, Piscataway, NJ, USA
- Tung-Dinh Pham** VNU University of Science, Hanoi, Vietnam
- Deepak Prajapati** Decision Sciences Area, Indian Institute of Management, Lucknow, India
- Yan Qin** School of Foreign Languages for Business, Southwestern University of Finance and Economics, Chengdu, China
- Alberto Regattieri** Department of Industrial Engineering, University of Bologna, Bologna, Italy
- Kodai Sugisaki** Tokyo City University, Tokyo, Japan
- Yoshinobu Tamura** Yamaguchi University, Yamaguchi, Japan
- Xiaolin Teng** Dotdash Meredith, New York, NY, USA
- Hoai An Le Thi** University of Lorraine, Metz, France
Institut Universitaire de France (IUF), Paris, France
- Kim Phuc Tran** University of Lille, ENSAIT, ULR 2461 - GEMTEX - Génie et Matériaux Textiles, Lille, France
- Phuong Hanh Tran** HEC-Liège, Management School of the University of Liège, Liège, Belgium
Univ. Lille, ENSAIT, ULR 2461 - GEMTEX - Genie et Materiaux Textiles, Lille, France
- Tzong-Ru Tsai** Department of Statistics, Tamkang University, New Taipei City, Taiwan
- Kwok-Leung Tsui** Grado Department of Industrial and Systems Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA
- Fugee Tsung** Department of Industrial Engineering and Decision Analytics, Hong Kong University of Science and Technology, Hong Kong, China
- Mai Phuong Vuong** Hanoi University of Science & Technology, Hanoi, Vietnam
- Kai Wang** School of Management, Xi'an Jiaotong University, Xi'an, Shaanxi, China
- Shuai Wang** School of Statistics, Southwestern University of Finance and Economics, Chengdu, China
- Xiao Wang** Purdue University, West Lafayette, IN, USA
- Yueyao Wang** Department of Statistics, Virginia Tech, Blacksburg, VA, USA
- Jing Wu** Department of Computer Science and Statistics, University of Rhode Island, Kingston, RI, USA
- Wei Biao Wu** University of Chicago, Chicago, IL, USA

Wen-Fang Wu Department of Mechanical Engineering, National Taiwan University, Taipei, Taiwan

Min-ge Xie Rutgers University, New Brunswick, NJ, USA

Min Xie Department of Systems Engineering and Engineering Management, City University of Hong Kong, Hong Kong, China

Yuhang Xu Bowling Green State University, Bowling Green, OH, USA

Shigeru Yamada Tottori University, Tottori, Japan

Jun Yan Department of Statistics, University of Connecticut, Storrs, CT, USA

Fangfang Yang School of Intelligent Systems Engineering Sun Yat-sen University, Guangdong, China

Hainan Zhang Department of Industrial & System Engineering, Rutgers University – New Brunswick, Piscataway, NJ, USA

Heping Zhang Department of Biostatistics, Yale University School of Public Health, New Haven, CT, USA

Xin Zhang Department of Statistics, Florida State University, Tallahassee, FL, USA

Zhiyang Zhang Department of Statistics, Virginia Tech, Blacksburg, VA, USA

Xufeng Zhao College of Economics and Management, Nanjing University of Aeronautics and Astronautics, Nanjing, China

Yang Zhao School of Public Health (Shenzhen), Sun Yat-sen University, Shenzhen, China

Xiaojun Zhu Xi'an Jiaotong-Liverpool University, Suzhou, China

Xi Zhu University of Washington, Seattle, WA, USA

Ruoqing Zhu University of Illinois at Urbana-Champaign, Champaign, IL, USA

Part I

Fundamental Statistics and Its Applications



Basic Statistics

1

Hoang Pham

Contents

1.1 Basic Probability Measures	3
1.1.1 Probability Axioms	4
1.1.2 Basic Statistics	4
1.1.3 Order Statistics	6
1.2 Reliability Measures	7
1.2.1 Coherent Systems	7
1.2.2 System Mean Time to Failure	9
1.2.3 Failure Rate Function	10
1.3 Common Probability Distribution Functions	11
1.3.1 Discrete Random Variable Distributions	11
1.3.2 Continuous Distributions	14
1.4 Statistical Inference and Estimation	22
1.4.1 Parameter Estimation	23
1.4.2 Maximum Likelihood Estimation with Censored Data	25
1.4.3 Statistical Change-Point Estimation Methods	28
1.4.4 Goodness-of-Fit Techniques	30
1.4.5 Least Squares Estimation	32
1.4.6 Interval Estimation	33
1.4.7 Nonparametric Tolerance Limits	35
1.4.8 Sequential Sampling	36
1.4.9 Bayesian Methods	37
1.5 Further Reading	37
Appendix A: Distribution Tables (Tables 1.7, 1.8, 1.9, 1.10, and 1.11)	37
References	41

Abstract

This chapter presents some fundamental elements of engineering probability and statistics with which some readers are probably already familiar, but others may not be. Statistics is the study of how best one can describe and analyze the data and then draw conclusions or inferences

based on the data available. The first section of this chapter begins with some basic definitions, including probability axioms, basic statistics, and reliability measures.

The second section describes the most common distribution functions such as the binomial, Poisson, geometric, exponential, normal, log normal, Student's t , gamma, Pareto, beta, Rayleigh, Cauchy, Weibull, Pham, and Vtub-shaped failure rate distributions, their applications, and their use in engineering and applied statistics.

The third section describes statistical inference, including parameter estimation and confidence intervals. Statistical inference is the process by which information from sample data is used to draw conclusions about the population from which the sample was selected that hopefully represents the whole population. This discussion also introduces the maximum likelihood estimation (MLE) method, the method of moments, MLE with censored data, the statistical change-point estimation method, nonparametric tolerance limits, sequential sampling, and Bayesian methods.

Finally, the last section provides a short list of books and articles for readers who are interested in advanced engineering and applied statistics.

Keywords

Failure rate · Parameter estimation · Distribution function · Interarrival time · Reliability function · Mean time to failure

1.1 Basic Probability Measures

We start off this chapter by defining several useful terms:

1. Outcome: A result or observation from an experiment, which cannot be predicted with certainty.
2. Event: Subset of a set of all possible outcomes.

H. Pham (✉)
Department of Industrial and Systems Engineering, Rutgers
University, Piscataway, NJ, USA
e-mail: hopham@soe.rutgers.edu

3. Probability: The relative frequency at which an event occurs in a large number of identical experiments.
4. Random variable: A function which assigns real numbers to the outcomes of an experiment.
5. Statistics: A function (itself a random variable) of one or more random variables that does not depend upon any unknown parameters.

1.1.1 Probability Axioms

Now let C be a subset of the sample space ($C \subset \Omega$). A probability set function, denoted by $P(C)$, has the following properties:

1. $P(\Omega) = 1, P(C) \geq 0$
2. $P(C_1 \cup C_2 \cup \dots) = P(C_1) + P(C_2) + \dots$

where the subsets C_i have no elements in common (i.e., they are mutually exclusive).

Let C_1 and C_2 be two subsets of the sample space Ω . The conditional probability of getting an outcome in C_2 given that an outcome from C_1 is given by

$$P(C_2|C_1) = \frac{P(C_2 \cap C_1)}{P(C_1)}.$$

Let C_1, C_2, \dots, C_n be n mutually disjoint subsets of the sample space Ω . Let C be a subset of the union of the C_i s, that is

$$C \subset \bigcup_{i=1}^n C_i.$$

Then

$$P(C) = \sum_{i=1}^n P(C|C_i)P(C_i) \quad (1.1)$$

and

$$P(C_i|C) = \frac{P(C|C_i)P(C_i)}{\sum_{i=1}^n P(C|C_i)P(C_i)}.$$

Equation (1.1) is known as the law of total probability or Bayes' rule.

Example 1.1 Suppose the total inventory of a company is a collection of lots from four different suppliers, A, B, C, and D, as indicated below:

Supplier	Percent of inventory
A	60
B	20
C	5
D	15

Furthermore, suppose that past records indicate that lots from suppliers A, B, C, and D are 5, 3, 2, and 8% defective, respectively. Using the Bayes' rule, the probability that a defective item selected as random is from supplier B can be calculated as follows:

$$P(B|\text{defective}) = \frac{(0.2)(0.03)}{(0.6)(0.05) + (0.2)(0.03) + (0.05)(0.02) + (0.15)(0.08)} = 0.122$$

1.1.2 Basic Statistics

The cumulative distribution function (cdf) F is a unique function which gives the probability that a random variable X takes on values less than or equal to some value x . In other word, $F(x) = P(X \leq x)$.

The probability density function (pdf) f is the probability that X takes on the value x ; that is, $f(x) = P(X = x)$.

The pdf satisfies the following two relations for discrete and continuous random variables, respectively:

$$\sum_{\text{all } x} f(x) = 1$$

and

$$\int_{-\infty}^{\infty} f(x)dx = 1.$$

In the continuous case, the pdf is the derivative of the cdf:

$$f(x) = \frac{\partial F(x)}{\partial x}.$$

The expected value of a random variable X is given by

$$E(X) = \sum_{\text{all } x} xf(x)$$

in the discrete case, and by

$$E(X) = \int_{-\infty}^{\infty} xf(x)dx$$

in the continuous case. Similarly, the variance of a random variable X , denoted by σ^2 , is a measure of how the values of X are spread about the mean value. It is defined as

$$\sigma^2 = E(X - \mu)^2.$$

It is calculated for discrete and continuous random variables, respectively, by

$$\sigma^2 = \sum_{\text{all } x} (x - \mu)^2 f(x)$$

and

$$\sigma^2 = \int_{-\infty}^{\infty} (x - \mu)^2 f(x) dx.$$

The standard deviation of X , denoted by σ , is the square root of the variance.

The skewness coefficient of a random variable X is a measure of the symmetry of the distribution of X about its mean value μ , and is defined as

$$S_c = \frac{E(X - \mu)^3}{\sigma^3}.$$

The skewness is zero for a symmetric distribution, negative for a left-tailed distribution, and positive for a right-tailed distribution.

Similarly, the kurtosis coefficient of a random variable X is a measure of how much of the mass of the distribution is contained in the tails, and is defined as

$$K_c = \frac{E(X - \mu)^4}{\sigma^4}.$$

Obviously, kurtosis is always positive; however, larger values represent distributions with heavier tails. The measure of kurtosis serves to differentiate between a flat distribution curve and a sharply peaked curve.

Given a random sample of size n from a distribution, the sample mean and sample variance are defined as, respectively,

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \quad \text{and} \quad S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2. \quad (1.2)$$

If the data are arranged in order of magnitude, the median is the central point of the series, i.e., there are equal numbers of observations greater than and less than the medians. In case if n (the total number of observations) is even, it is usual to take the mean of the two central values as the median.

Suppose that n values of a variable X are ordered in increasing order of magnitude, i.e., we have x_1, x_2, \dots, x_n such that $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$ then the median of a set of values is equal to $x_{(\frac{n+1}{2})}$ if n is odd and $\frac{1}{2} (x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)})$ if n is even.

The median of a random variable X is the value m such that the cdf $F(m) = \frac{1}{2}$. The median always exists if the random variable is continuous, whereas it may not always exist if the random variable is discrete.

Example 1.2 The probability density function of a random variable X is

$$f(x) = \begin{cases} 2x & 0 \leq x \leq 1 \\ 0 & \text{otherwise.} \end{cases}$$

The cdf of X is given by

$$F(x) = \begin{cases} 0 & x < 0 \\ x^2 & 0 \leq x \leq 1 \\ 1 & x > 1. \end{cases}$$

Set $F(m) = 1/2$, then the median of X is $\frac{\sqrt{2}}{2}$.

The mode is the value of the variate which occurs most frequently for which the frequency is a maximum. In other words, the mode of a set of values of a variable is the most frequently occurring value in the set. The use of the mode is most often associated with discrete distributions where it is simple to count which value of the variate occurs most often.

Example 1.3 The density function of X is given by

$$f(x) = \begin{cases} \frac{2}{3}x & 0 \leq x \leq 1 \\ \frac{1}{3} & 1 < x \leq 3. \end{cases}$$

The cdf of X is

$$F(x) = \begin{cases} 0 & x < 0 \\ \frac{x^2}{3} & 0 \leq x \leq 1 \\ \frac{x}{3} & 1 < x \leq 3 \\ 1 & x > 3. \end{cases}$$

Since $F(\frac{3}{2}) = \frac{1}{2}$, so the median of X is equal to $\frac{3}{2}$. Similarly, the mode is equal to 1 since $f(x)$ is a maximum when $x = 1$.

Which measure to choose? The mode should be used when calculating measure of center for the qualitative variable. When the variable is quantitative with symmetric distribution, then the mean is proper measure of center. In a case of quantitative variable with skewed distribution, the median is good choice for the measure of center. This is related to the fact that the mean can be highly influenced by an observation that falls far from the rest of the data, called an outlier.

Moment Inequalities

The following moment inequalities are commonly used in applied engineering statistics.

Theorem 1.1 (Jensen's Inequality) For any random variable X and a convex function $g(x)$,

$$g(E(X)) \leq E[g(X)]. \quad (1.3)$$

Example 1.4 Since the function $g(x) = x^2$ is a convex function, using Jensen's inequality we obtain

$$g[E(X)] \leq E(g(X))$$

or

$$[E(X)]^2 \leq E(X^2)$$

In other words, since $\text{var}(X) = E(X^2) - [E(X)]^2 \geq 0$ this implies that $E(X^2) \geq [E(X)]^2$.

Theorem 1.2 (Cauchy-Schwarz Inequality) For any two random variables X and Y ,

$$E|XY| \leq \sqrt{E(X^2)E(Y^2)}. \quad (1.4)$$

Theorem 1.3 (Holder's Inequality) For any two random variables X and Y and two positive real values a and b such that $a + b = 1$,

$$E|XY| \leq \sqrt{\left(E(|X|^{1/a})\right)^a \left(E(|Y|^{1/b})\right)^b}.$$

The conditional distribution of a random variable Y given that another random variable X takes on a value x is given by:

$$f(y|X=x) = \frac{f(x,y)}{f_1(x)},$$

where

$$f_1(x) = \int_{-\infty}^{\infty} f(x,y) dy.$$

Let X and Y be two independent lifetimes with pdf $f_1(x)$ and $f_2(y)$, respectively. Let $Z = X + Y$ be the total lifetime having the pdf $g(z)$ and cdf $G(z)$. Using a convolution approach, the pdf of z can be obtained as

$$\begin{aligned} g(z) &= \int_0^z f_1(x)f_2(z-x) dx = \int_0^z f_1(z-y)f_2(y)dy \\ &= f_1(z) * f_2(z) \end{aligned}$$

where $*$ denotes the convolutions of the distributions. Similarly, the distribution function of Z , $G(z)$, is given by

$$G(z) = P[Z \leq z] = \int_0^z f_1(x)dx \int_0^{z-x} f_2(y)dy.$$

Note that the two random variables X and Y are said to be independent if knowledge of one of the variables has no effect on the distribution of probability for the other.

1.1.3 Order Statistics

Let X_1, X_2, \dots, X_n be a random sample of size n , each with cdf $F(x)$ and pdf $f(x)$. Let $X_{(i)}$ be the i th order statistic of the sample where such characteristic values of a sample of observations have been sorted from smallest to largest. The r th order statistic is the r th smallest value in the sample, say $X_{(r)}$. That is,

$$X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(r-1)} \leq X_{(r)} \leq \dots \leq X_{(n)}$$

are called the *order statistics*. Imagine we are building an elevator, for example, and our random variables are the weight on the elevator of various times during the day. We would want to know how much weight it can bear. In this case, we will care about the distribution of the maximum.

Let $F_{(r)}(x)$ denote the cdf of the r th order statistic $X_{(r)}$. Note that the order statistics $X_{(1)}, X_{(2)}, \dots, X_{(n)}$ are neither independent nor identically distributed. Let $X_{(r)}$ be the r th order statistic. Then the cdf of $X_{(r)}$ is given by

$$\begin{aligned} F_{(r)}(x) &= P(X_{(r)} \leq x) \\ &= P(\text{at least } r \text{ of } X_1, X_2, \dots, X_n \text{ are } \leq x) \\ &= \sum_{i=r}^n P(\text{exactly } i \text{ of } X_1, X_2, \dots, X_n \text{ are } \leq x) \\ &= \sum_{i=r}^n \binom{n}{i} F^i(x)[1-F(x)]^{n-i} \\ &= F^r(x) \sum_{j=0}^{n-r} \binom{r+j-1}{r-1} [1-F(x)]^j \\ &= r \binom{n}{r} \int_0^{F(x)} t^{r-1} (1-t)^{n-r} dt \end{aligned}$$

The probability density function (pdf) of the r th order statistic $X_{(r)}$ is given by

$$f_{X_{(r)}}(x) = \frac{n!}{(r-1)!(n-r)!} f(x) F^{r-1}(x) [1-F(x)]^{n-r}$$

The pdf of the largest, i.e., n st order statistic $X_{(n)}$, value is given by

$$F_{X_{(n)}}(x) = nf(x)[F(x)]^{n-1}$$

Similarly, the pdf of the smallest, i.e., first-order statistic $X_{(1)}$, value of n independent random variables, is given by

$$F_{X_{(1)}}(x) = nf(x)[1 - F(x)]^{n-1}$$

Assume there are n random variables X_1, X_2, \dots, X_n which may or may not be mutually independent. The joint cdf, if it exists, is given by

$$P(X_1 \leq x_1, X_2 \leq x_2, \dots, X_n \leq x_n) = \int_{-\infty}^{x_n} \int_{-\infty}^{x_{n-1}} \dots \int_{-\infty}^{x_1} f(t_1, t_2, \dots, t_n) dt_1 dt_2 \dots dt_n \tag{1.5}$$

If the n random variables are mutually statistically independent, then the joint pdf can be rewritten as:

$$f(x_1, x_2, \dots, x_n) = \prod_{i=1}^n f(x_i).$$

1.2 Reliability Measures

1.2.1 Coherent Systems

In reliability theory, both the system and its components are commonly allowed to take only two possible states or binary states: either working or failed. In a multistate system, both the system and its components are allowed to experience more than two possible states, e.g., fully working, partially or degraded working, or failed. A multistate system reliability model provides more flexibility for modeling of system's conditions. The reliability can be defined for a more general coherent structure. A coherent system with binary states can be simply described as follows. Consider that a system consists of n components and define

$$x_i = \begin{cases} 1 & \text{if component } i \text{ is functioning} \\ 0 & \text{if component } i \text{ is failed} \end{cases} \quad \text{for } i = 1, 2, \dots, n$$

and

$$\phi(\mathbf{x}) = \begin{cases} 1 & \text{if the system is functioning} \\ 0 & \text{if the system is failed} \end{cases}$$

where $\mathbf{x} = (x_1, x_2, \dots, x_n)$. Binary indicators x_i and $\phi(\mathbf{x})$ indicate the state of component i and the state of the system, respectively. The function $\phi(\mathbf{x})$ is referred to as the structure function of the system. Define

$$\begin{aligned} (0_i, \mathbf{x}) &= (x_1, x_2, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \\ (1, \mathbf{x}) &= (x_1, x_2, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n). \end{aligned}$$

The *structure function* of the system indicates that the state of the system is completely determined by the states of all components. A component is relevant if its state does affect the state of the system.

Definition 1.1 A binary system with n components is said to be coherent if:

- The structure function ϕ is nondecreasing in each argument $x_i, i = 1, 2, \dots, n$
- Each component is relevant, i.e., there exists at least one vector \mathbf{x} such that $\phi(1_i, \mathbf{x}) = 1$ and $\phi(0_i, \mathbf{x}) = 0$
- $\phi(\mathbf{0}) = 0$ and $\phi(\mathbf{1}) = 1$

For coherent systems, when $\mathbf{x} < \mathbf{y}$ then $\phi(\mathbf{x}) \leq \phi(\mathbf{y})$.

Example 1.4 For a series system consisting of n components to function, all its components must function. Then structure function of the series system is given by

$$\phi(\mathbf{x}) = \prod_{i=1}^n x_i = \min(x_1, x_2, \dots, x_n)$$

For the parallel system consisting of n components to work there is at least one component to work. Then structure function of the parallel system is given by

$$\phi(\mathbf{x}) = 1 - \prod_{i=1}^n (1 - x_i) = \max(x_1, x_2, \dots, x_n)$$

Similarly, the k out of n system is functioning if and only if there is at least k component to function. Then structure function of the k out of n system is given by

$$\phi(\mathbf{x}) = \begin{cases} 1 & \text{if } \sum_{i=1}^n x_i \geq k \\ 0 & \text{if } \sum_{i=1}^n x_i < k \end{cases}$$

We now discuss the minimal path and minimal cut sets. Let A_1, A_2, \dots, A_s denote the minimal path sets of a given system.

Definition 1.2 A state vector \mathbf{x} is called a *path vector* (or link vector) if $\phi(\mathbf{x}) = 1$.

A path set is a set of components whose functioning ensures the functioning of the system.

Definition 1.3 A *minimal path vector* is a path vector for which the failure of any functioning components results in system failure.

Definition 1.4 A *minimal path set* is a minimal set of components whose functioning ensures the functioning of the system.

In other words, a *minimal path vector* is a path vector \mathbf{x} such that $\phi(\mathbf{y}) = 0$ for any $\mathbf{y} < \mathbf{x}$ and its corresponding path set is *minimal path set*. Let

$$\alpha_j(\mathbf{x}) = \begin{cases} 1 & \text{iff all the components of } A_j \text{ are functioning} \\ 0 & \text{otherwise} \end{cases}$$

then

$$\alpha_j(\mathbf{x}) = \prod_{i \in A_j} X_i$$

A system will function if and only if all the components of at least one minimal path sets are functioning. That is,

$$\begin{aligned} \phi(\mathbf{x}) &= \max_j \alpha_j(\mathbf{x}) \\ &= 1 - \prod_{j=1}^s (1 - \alpha_j(\mathbf{x})) \\ &= 1 - \prod_{j=1}^s \left(1 - \prod_{i \in A_j} X_i \right). \end{aligned} \quad (1.6)$$

Similarly, let C_1, C_2, \dots, C_k denote the minimal cut sets of a given system.

Definition 1.5 A state vector \mathbf{x} is called a *cut vector* if $\phi(\mathbf{x}) = 0$. A cut set is a set of components whose failure ensures the failure of the system.

Definition 1.6 A *minimal cut vector* is a cut vector for which the repair of any failed component results in a functioning system.

Definition 1.7 A *minimal cut set* is a minimal set of components whose failure ensures the failure of the system.

In other words, a *minimal cut vector* is a cut vector \mathbf{x} such that $\phi(\mathbf{y}) = 1$ for any $\mathbf{y} > \mathbf{x}$ and its corresponding cut set is *minimal cut set*.

Note that for a minimal path set to work, each component in the set must work. For a minimal cut set to fail, all components in the set must fail.

Let

$$\beta_j(\mathbf{x}) = \begin{cases} 1 & \text{if at least one component in} \\ & \text{the } j\text{th minimal cut set is functioning} \\ 0 & \text{otherwise for } j = 1, 2, \dots, k \end{cases}$$

then

$$\beta_j(\mathbf{x}) = \max_{i \in C_j} X_i$$

A system is not functioning (or system unreliability) if and only if all the components of at least one minimal cut set are not functioning. In other words, a system is functioning if and only if at least one component in each minimal cut sets is functioning. Thus, the structure function based on the minimal cut sets is given by

$$\begin{aligned} \phi(\mathbf{x}) &= \prod_{j=1}^k \beta_j(\mathbf{x}) \\ &= \prod_{j=1}^k \max_{i \in C_j} X_i \\ &= \prod_{j=1}^k \left(\prod_{i \in C_j} X_i \right) \\ &= \prod_{j=1}^k \left(1 - \prod_{i \in C_j} (1 - X_i) \right). \end{aligned}$$

Let p_i is the component i reliability. That is,

$$p_i = P(X_i = 1) = E[X_i].$$

The system reliability function can be defined as

$$r(p) = P(\phi(\mathbf{x}) = 1) = E[\phi(\mathbf{x})].$$

Theorem 1.4 If ϕ is a coherent system of independent components with minimal path set A_1, A_2, \dots, A_s and minimal cut sets C_1, C_2, \dots, C_k then

$$\prod_{j=1}^k \left(1 - \prod_{i \in C_j} (1 - p_i) \right) \leq r(p) \leq 1 - \prod_{j=1}^s \left(1 - \prod_{i \in A_j} p_i \right). \quad (1.7)$$

Example 1.5 A k -out-of- n system works if and only if at least k of the n components work. In a k -out-of- n system, there are $\binom{n}{k}$ minimal path sets and $\binom{n}{n-k+1}$ minimal cut sets. Each minimal path set contains exactly k different components and each minimal cut set contains exactly $n - k + 1$ components. Thus, all minimal path sets and minimal cut sets are known. One can easily obtain the reliability of a k -out-of- n system either in terms of the minimal path sets or minimal cut sets.

Definitions of reliability given in the literature vary according to the practitioner or researcher. The generally accepted definition is as follows:

Definition 1.8 *Reliability* is the probability that the system will perform its intended function under specified design limits.

More specifically, reliability is the probability that a product or system will operate properly for a specified period of time (design life) under the design operating conditions (such as temperature, voltage, etc.) without failure. In other words, reliability can be used as a measure of the system's success at providing its function properly. Reliability is one of the quality characteristics that consumers require from manufacturers.

Mathematically, reliability $R(t)$ is the probability that a system will be successful in the interval from time 0 to time t :

$$R(t) = P(T > t), \quad t \geq 0, \quad (1.8)$$

where T is a random variable denoting the time to failure or failure time.

Unreliability, or the cdf $F(t)$, a measure of failure, is defined as the probability that the system will fail by time t .

$$F(t) = P(T \leq t), \quad t \geq 0.$$

In other words, $F(t)$ is the failure distribution function. If the time-to-failure random variable T has a density function $f(t)$, then

$$R(t) = \int_t^{\infty} f(s) ds$$

or, equivalently,

$$f(t) = -\frac{d}{dt} [R(t)].$$

The density function can be mathematically described in terms of T :

$$\lim_{\Delta t \rightarrow 0} P(t < T \leq t + \Delta t).$$

This can be interpreted as the probability that the failure time T will occur between the operating time t and the next interval of operation $t + \Delta t$.

Consider a new and successfully tested system that operates well when put into service at time $t = 0$. The system becomes less likely to remain successful as the time interval increases. The probability of success for an infinite time interval is, of course, zero. Thus, the system starts to function at a probability of one and eventually decreases to a probability of zero. Clearly, reliability is a function of mission time. For example, one can say that the reliability of the system is 0.995 for a mission time of 24 h.

Example 1.6 A computer system has an exponential failure time density function

$$f(t) = \frac{1}{9000} e^{-\frac{t}{9000}}, \quad t \geq 0.$$

The probability that the system will fail after the warranty (six months or 4380 h) and before the end of the first year (one year or 8760 h) is given by

$$\begin{aligned} P(4380 < T \leq 8760) &= \int_{4380}^{8760} \frac{1}{9000} e^{-\frac{t}{9000}} dt \\ &= 0.237. \end{aligned}$$

This indicates that the probability of failure during the interval from 6 months to 1 year is 23.7%.

Consider the Weibull distribution, where the failure time density function is given by

$$f(t) = \frac{\beta t^{\beta-1}}{\theta^{\beta}} e^{-(\frac{t}{\theta})^{\beta}}, \quad t \geq 0, \theta > 0, \beta > 0.$$

Then the reliability function is

$$R(t) = e^{-(\frac{t}{\theta})^{\beta}}, \quad t \geq 0.$$

Thus, given a particular failure time density function or failure time distribution function, the reliability function can be obtained directly. Section 1.2 provides further insight for specific distributions.

Definition 1.9 *Conditional reliability* is the probability of surviving a mission of length h under specified design limits given that the system has been survived up to time t .

In other words, conditional reliability $R(h|t)$ is the probability that a system will be successful in the interval $(t, t + h)$ given that it already survived more than t . Mathematically,

$$R(h|t) = P(T > t + h | T > t) = \frac{R(t+h)}{R(t)} \quad t \geq 0, h > 0$$

It is obvious that the probability of prolongation by additional mission length h decreases with increasing h . Indeed, the longer prolongation of life time required, the smaller is the probability of such an event.

1.2.2 System Mean Time to Failure

Suppose that the reliability function for a system is given by $R(t)$. The expected failure time during which a component is expected to perform successfully, or the system mean time to failure (MTTF), is given by

$$\text{MTTF} = \int_0^{\infty} tf(t)dt$$

or, equivalently, that

$$\text{MTTF} = \int_0^{\infty} R(t)dt. \quad (1.9)$$

Thus, MTTF is the definite integral evaluation of the reliability function. In general, if $\lambda(t)$ is defined as the failure rate function, then, by definition, MTTF is not equal to $1/\lambda(t)$.

The MTTF should be used when the failure time distribution function is specified because the reliability level implied by the MTTF depends on the underlying failure time distribution. Although the MTTF measure is one of the most widely used reliability calculations, it is also one of the most misused calculations. It has been misinterpreted as a “guaranteed minimum lifetime.” Consider the results given in Table 1.1 for a 12-component life duration test.

A component MTTF of 3660 h was estimated using a basic averaging technique. However, one of the components failed after 920 h. Therefore, it is important to note that the system MTTF denotes the average time to failure. It is neither the failure time that could be expected 50% of the time nor is it the guaranteed minimum time of system failure, but mostly depends on the failure distribution.

A careful examination of Eq. (1.4) will show that two failure distributions can have the same MTTF and yet produce different reliability levels.

Example 1.7 Assume that the failure rate for a hydraulic component is given by

$$h(t) = \frac{t}{t+1} \quad t > 0$$

Table 1.1 Results from a 12-component life duration test

Component	Time to failure (h)
1	4510
2	3690
3	3550
4	5280
5	2595
6	3690
7	920
8	3890
9	4320
10	4770
11	3955
12	2750

where t is in years. Note that the reliability function $R(t)$ can be written as follows:

$$R(t) = e^{-\int_0^t h(x)dx}.$$

Then we obtain

$$R(t) = e^{-\int_0^t \frac{x}{x+1} dx} = (t+1)e^{-t}$$

From Eq. (1.9), the MTTF of the component is given by

$$\text{MTTF} = \int_0^{\infty} R(t)dt = \int_0^{\infty} (t+1)e^{-t} dt = 2$$

1.2.3 Failure Rate Function

The probability of a system failure in a given time interval $[t_1, t_2]$ can be expressed in terms of the reliability function as

$$\begin{aligned} \int_{t_1}^{t_2} f(t)dt &= \int_{t_1}^{\infty} f(t)dt - \int_{t_2}^{\infty} f(t)dt \\ &= R(t_1) - R(t_2) \end{aligned}$$

or in terms of the failure distribution function (or the unreliability function) as

$$\begin{aligned} \int_{t_1}^{t_2} f(t)dt &= \int_{-\infty}^{t_2} f(t)dt - \int_{-\infty}^{t_1} f(t)dt \\ &= F(t_2) - F(t_1). \end{aligned}$$

The rate at which failures occur in a certain time interval $[t_1, t_2]$ is called the *failure rate*. It is defined as the probability that a failure per unit time occurs in the interval, given that a failure has not occurred prior to t_1 , the beginning of the interval. Thus, the failure rate is

$$\frac{R(t_1) - R(t_2)}{(t_2 - t_1)R(t_1)}.$$

Note that the failure rate is a function of time. If we redefine the interval as $[t, t + \Delta t]$, the above expression becomes

$$\frac{R(t) - R(t + \Delta t)}{\Delta t R(t)}.$$

The rate in the above definition is expressed in failures per unit time, but in reality the time units might instead correspond to miles, hours, trials, etc. The *hazard function* is defined as the limit of the failure rate as the interval

approaches zero. Thus, the hazard function $h(t)$ is the instantaneous failure rate, and is defined by

$$\begin{aligned} h(t) &= \lim_{\Delta t \rightarrow 0} \frac{R(t) - R(t + \Delta t)}{\Delta t R(t)} \\ &= \frac{1}{R(t)} \left[-\frac{d}{dt} R(t) \right] \\ &= \frac{f(t)}{R(t)}. \end{aligned} \quad (1.10)$$

The quantity $h(t)dt$ represents the probability that a device of age t will fail in the small interval of time t to $(t + dt)$. The importance of the hazard function is that it indicates the change in the failure rate over the life of a population of components by plotting their hazard functions on a single axis. For example, two designs may provide the same reliability at a specific point in time, but the failure rates up to this point in time can differ.

The death rate, in statistical theory, is analogous to the failure rate, as the nature of mortality is analogous to the hazard function. Therefore, the hazard function, hazard rate, or failure rate function is the ratio of the pdf to the reliability function.

Example 1.8 Consider the lifetime of a communication device, having a pdf

$$f(t) = \begin{cases} 0 & t \leq t_0 \\ \frac{4(t-t_0)}{(t_1-t_0)^2} & t_0 \leq t \leq \frac{t_0+t_1}{2} \\ \frac{4(t_1-t)}{(t_1-t_0)^2} & \frac{t_0+t_1}{2} \leq t \leq t_1 \\ 0 & t_1 \leq t \end{cases}$$

Then the cdf is

$$F(t) = \begin{cases} 0 & t \leq t_0 \\ \frac{2(t-t_0)^2}{(t_1-t_0)^2} & t_0 \leq t \leq \frac{t_0+t_1}{2} \\ 1 - \frac{2(t_1-t)^2}{(t_1-t_0)^2} & \frac{t_0+t_1}{2} \leq t \leq t_1 \\ 1 & t_1 \leq t \end{cases}$$

Mean Residual Life Function

Another important measure of reliability is the mean residual life $\mu(t)$, which is the expected remaining life beyond the present age t . Mathematically, the mean residual life can be defined as

$$\mu(t) = E[T - t | T \geq t] = \frac{\int_t^\infty R(s) ds}{R(t)}. \quad (1.11)$$

In other words, the mean residual life (MRL) is the expected remaining life, $T - t$, given that the item has survived to time t . When $t = 0$, the MRL function becomes the MTTF.

1.3 Common Probability Distribution Functions

In some situations, particularly when evaluating reliability of a product in the field, exact information on the failure times is not possible to obtain, but the information on the number of failures during the time interval $[a, b]$ perhaps is available. The difference between the actual time til failure, T , and the number of failures, N , is that T is a continuous random variable while N is a discrete random variable. Random variables can be discrete or continuous.

This section presents some of the common discrete and continuous distribution functions and several hazard models that are applied in engineering statistics [1].

1.3.1 Discrete Random Variable Distributions

Binomial Distribution

The binomial distribution is one of the most widely used discrete random variable distributions in reliability and quality inspection. It has applications in reliability engineering, for example, when one is dealing with a situation in which an event is either a success or a failure.

The binomial distribution can be used to model a random variable X which represents the number of successes (or failures) in n independent trials (these are referred to as Bernoulli trials), with the probability of success (or failure) being p in each trial. The pdf of the distribution is given by

$$P(X = x) = \binom{n}{x} p^x (1-p)^{n-x}, \quad x = 0, 1, 2, \dots, n,$$

$$\binom{n}{x} = \frac{n!}{x!(n-x)!},$$

where n = number of trials, x = number of successes, and p = single trial probability of success.

The mean of the binomial distribution is np and the variance is $np(1-p)$. The coefficient of skewness is given by

$$S_c = \frac{1-2p}{\sqrt{np(1-p)}}$$

and the coefficient of kurtosis is

$$K_c = 3 - \frac{6}{n} + \frac{1}{np(1-p)}.$$

The reliability function $R(k)$ (i.e., at least k out of n items are good) is given by

$$R(k) = \sum_{x=k}^n \binom{n}{x} p^x (1-p)^{n-x}.$$

Example 1.9 Suppose that, during the production of lightbulbs, 90% are found to be good. In a random sample of 20 lightbulbs, the probability of obtaining at least 18 good lightbulbs is given by

$$\begin{aligned} R(18) &= \sum_{x=18}^{20} \binom{20}{x} (0.9)^x (0.1)^{20-x} \\ &= 0.667. \end{aligned}$$

Poisson Distribution

Although the Poisson distribution can be used in a manner similar to the binomial distribution, it is used to deal with events in which the sample size is unknown. A Poisson random variable is a discrete random variable distribution with a probability density function given by

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!} \quad \text{for } x = 0, 1, 2, \dots \quad (1.12)$$

where $\lambda =$ constant failure rate and $x =$ the number of events. In other words, $P(X = x)$ is the probability that exactly x failures occur.

A Poisson distribution is used to model a Poisson process. A Poisson random variable has a mean and a variance both equal to λ where λ is called the parameter of the distribution. The skewness coefficient is

$$S_c = \frac{1}{\sqrt{\lambda}}$$

and the kurtosis coefficient is

$$K_c = 3 + \frac{1}{\lambda}.$$

The Poisson distribution reliability up to time t , $R(k)$ (the probability of k or fewer failures), can be defined as follows:

$$R(k) = \sum_{x=0}^k \frac{(\lambda t)^x e^{-\lambda t}}{x!}.$$

This distribution can be used to determine the number of spares required for a system during a given mission.

Example 1.10 A nuclear plant is located in an area susceptible to both high winds and earthquakes. From historical data, the mean frequency of large earthquakes capable of damaging important plant structures is one every 50 years. The corresponding frequency of damaging high winds is

once in 25 years. During a strong earthquake, the probability of structure damage is 0.1. During high winds, the damage probability is 0.05. Assume that earthquakes and high winds can be described by independent Poisson random variables and that the damage caused by these events are independent. Let us answer the following questions:

1. What is the probability of having strong winds but not large earthquakes during a ten-year period?
2. What is the probability of having strong winds and large earthquakes in the ten-year period?
3. What is the probability of building damage during the ten-year period?

Considering the first question, let the random variables X and Y represent the number of earthquakes and the number of occurrences of high winds, respectively. We assume that the two random variables are statistically independent. The means of X and Y are, respectively, given by

$$\lambda_X = \frac{1}{50 \text{ y}} (10 \text{ y}) = 0.2$$

and

$$\lambda_Y = \frac{1}{25 \text{ y}} (10 \text{ y}) = 0.4.$$

The conditional damage probabilities are given as follows:

$$P(\text{damage/earthquake}) = 0.1$$

and

$$P(\text{damage/wind}) = 0.05.$$

Let event

- $A = \{\text{strong winds and no earthquakes}\}$
- $B = \{\text{strong winds and large earthquakes}\}$
- $C = \{\text{building damage}\}$

Assuming that the winds and earthquakes are independent of each other, the probability of having strong winds but not earthquakes during the ten-year period can be written as:

$$\begin{aligned} P(A) &= P(\text{winds}) P(\text{no earthquakes}) \\ &= [1 - P(\text{no winds})] P(\text{no earthquakes}) \end{aligned}$$

Therefore, we obtain

$$P(A) = (1 - e^{-0.4}) (e^{-0.2}) = 0.27$$

For the second question, the probability of having strong winds and earthquakes during the ten-year period can be obtained from

$$\begin{aligned} P(B) &= P(\text{winds}) P(\text{earthquakes}) \\ &= [1 - P(\text{no winds})] [1 - P(\text{no earthquakes})] \\ &= (1 - e^{-0.4}) (1 - e^{-0.2}) = 0.06. \end{aligned}$$

Finally, for the third question, we assume that multiple occurrences of earthquakes and high winds do not occur during the ten-year period. Therefore, the probability of building damage can be written as

$$\begin{aligned} P(C) &= P(\text{damage/earthquakes}) P(\text{earthquakes}) \\ &\quad + P(\text{damage/wind}) P(\text{wind}) \\ &\quad - P(\text{damage/earthquakes and wind}) \\ &\quad P(\text{earthquake and wind}) \\ &= P(\text{damage/earthquakes}) P(\text{earthquakes}) \\ &\quad + P(\text{damage/wind}) P(\text{wind}) \\ &\quad - P(\text{damage/earthquakes}) P(\text{damage/wind}) \\ &\quad P(\text{earthquake and wind}) \\ &= (1 - e^{-0.2}) (0.1) + (1 - e^{-0.4}) (0.05) \\ &\quad - (0.05)(0.1)(0.06) \\ &= 0.0343. \end{aligned}$$

Geometric Distribution

Consider a sequence of independent trials where each trial has the same probability of success, p . Let N be a random variable representing the number of trials until the first success. This distribution is called the geometric distribution. It has a pdf given by

$$P(N = n) = p(1 - p)^{n-1}, \quad n = 1, 2, \dots$$

The corresponding cdf is

$$F(n) = 1 - (1 - p)^n, \quad n = 1, 2, \dots$$

The expected value and the variance are, respectively,

$$E(N) = \frac{1}{p}$$

and

$$V(N) = \frac{1 - p}{p^2}.$$

It should be noted that the geometric distribution is the only discrete distribution having the *memoryless property*. In other words, mathematically, it is

$$P(N \geq (n + s) | N > s) = P(N \geq n) \quad n = 1, 2, \dots$$

This memoryless property can be shown as follows. We know that

$$P(N > n) = P(N \geq n + 1) = (1 - p)^n$$

Thus,

$$\begin{aligned} P(N \geq (n + s) | N > s) &= \frac{P(N \geq n + s)}{P(N > s)} \\ &= \frac{(1 - p)^{n+s-1}}{(1 - p)^s} \\ &= (1 - p)^{n-1} \\ &= P(N \geq n). \end{aligned}$$

The Pascal Distribution

The random variable X for the Pascal distribution represents the number of trials until the r th success where p is the probability of success of each trial. The Pascal distribution of X is given by

$$P(X = n) = \binom{n-1}{r-1} p^r (1-p)^{n-r} \quad n = r, r+1, r+2, \dots \quad (1.13)$$

The Pascal distribution is the probability distribution of a certain number of successes and failures in a series of independent and identically distributed Bernoulli trials. For n Bernoulli trials with success probability p , the Pascal distribution gives the probability of r successes and $(n - r)$ failures, with a success on the last trial. The mean and the variance of the Pascal distribution are

$$E(X) = \frac{r}{p}$$

and

$$V(X) = \frac{r(1-p)}{p^2}$$

respectively. When $r = 1$ the Pascal distribution becomes geometric distribution.

Hypergeometric Distribution

The hypergeometric distribution is a discrete distribution that arises in sampling, for example. It has a pdf given by

$$f(x) = \frac{\binom{k}{x} \binom{N-k}{n-x}}{\binom{N}{n}} \quad x = 0, 1, 2, \dots, n. \quad (1.14)$$

Typically, N will be the number of units in a finite population; n will be the number of samples drawn without replacement from N ; k will be the number of failures in the population; and x will be the number of failures in the sample.

The expected value and variance of the hypergeometric random variable X are, respectively

$$E(X) = \frac{nk}{N}$$

and

$$V(X) = \frac{k(N-k)n(N-n)}{N^2(N-1)}.$$

1.3.2 Continuous Distributions

Exponential Distribution

The exponential distribution plays an essential role in reliability engineering because it has a constant failure rate. It has been used to model the lifetimes of electronic and electrical components and systems. This distribution is applicable to the case where a used component that has not failed is as good as a new component – a rather restrictive assumption. It should therefore be used carefully, since there are numerous situations where this assumption (known as the “memoryless property” of the distribution) is not valid.

If the time to failure is described by an exponential failure time density function, then

$$f(t) = \frac{1}{\theta} e^{-\frac{t}{\theta}}, \quad t \geq 0, \theta > 0 \quad (1.15)$$

and this will lead to the reliability function

$$R(t) = \int_t^{\infty} \frac{1}{\theta} e^{-\frac{s}{\theta}} ds = e^{-\frac{t}{\theta}}, \quad t \geq 0,$$

where $\theta = 1/\lambda > 0$ is an MTTF's parameter and $\lambda \geq 0$ is a constant failure rate. Figure 1.1 shows the exponential pdf for various values of λ .

The hazard function or failure rate for the exponential density function is constant, i.e.,

$$h(t) = \frac{f(t)}{R(t)} = \frac{\frac{1}{\theta} e^{-\frac{t}{\theta}}}{e^{-\frac{t}{\theta}}} = \frac{1}{\theta} = \lambda.$$

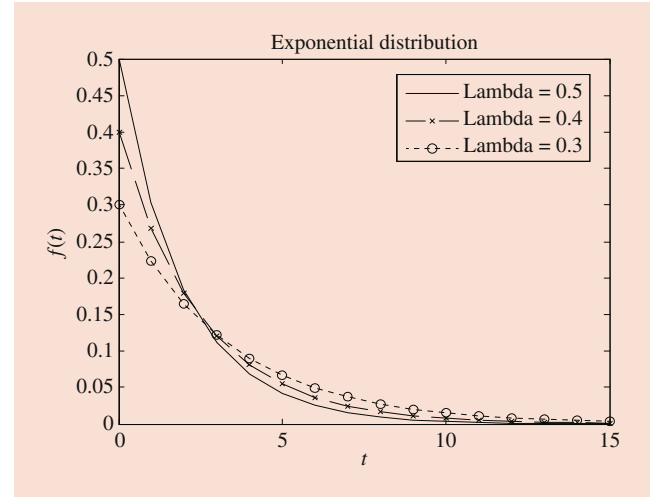


Fig. 1.1 Exponential density function for various values of λ

The failure rate for this distribution is λ , a constant, which is the main reason for this widely used distribution. Because of its constant failure rate, the exponential is an excellent model for the long flat “intrinsic failure” portion of the bathtub curve. Since most parts and systems spend most of their lifetimes in this portion of the bathtub curve, this justifies frequent use of the exponential distribution (when early failures or wearout is not a concern). The exponential model works well for interarrival times. When these events trigger failures, the exponential lifetime model can be used.

We will now discuss some properties of the exponential distribution that can be used to understand its characteristics and when and where it can be applied.

Property 1.1 (*Memoryless property*). The exponential distribution is the only continuous distribution that satisfies

$$P\{T \geq t\} = P\{T \geq t + s | T \geq s\} \quad \text{for } t > 0, s > 0.$$

This result indicates that the conditional reliability function for the lifetime of a component that has survived to time s is identical to that of a new component. This term is the so-called “used as good as new” assumption.

Property 1.2 If T_1, T_2, \dots, T_n are independently and identically distributed exponential random variables (r.v.'s) with a constant failure rate λ , then

$$2\lambda \sum_{i=1}^n T_i \sim \chi^2(2n), \quad (1.16)$$

where $\chi^2(2n)$ is a chi-squared distribution with $2n$ degrees of freedom. This result is useful for establishing a confidence interval for λ .

Lower-Bound Truncated Exponential Distribution

The lower-bound truncated exponential distribution is an exponential distribution starting at t_0 . The model can be used when no unit fails before the time t_0 . The pdf of a lower-bound truncated exponential function can be defined as follows:

$$f(t) = \begin{cases} 0 & t < t_0 \\ \frac{1}{\theta} e^{-\frac{t-t_0}{\theta}} & t \geq t_0 \end{cases}$$

The parameters θ and t_0 are the scale parameter and location parameter. Note that the failure rate of this lower-bound truncated exponential distribution is a constant for all $t \geq t_0$. This can be easily obtained as follows. The reliability function is given by

$$R(t) = \int_t^{\infty} \frac{1}{\theta} e^{-\frac{x-t_0}{\theta}} dx = e^{-\frac{t-t_0}{\theta}}$$

Thus, the failure rate of truncated exponential distribution is

$$h(t) = \frac{f(t)}{R(t)} = \frac{1}{\theta}.$$

Upper-Bound Truncated Exponential Distribution

For many applications, it is desired to consider to truncate a distribution on an upper tail. Let us assume that the exponential density be truncated at time T . The pdf of an upper-bound truncated exponential distribution function is defined as

$$f(t) = \begin{cases} 0 & t > T \\ \frac{\lambda e^{-\lambda t}}{1 - e^{-\lambda T}} & 0 < t \leq T \end{cases}$$

The reliability function is

$$R(t) = \int_t^T f(x) dx = \int_t^T \frac{\lambda e^{-\lambda x}}{1 - e^{-\lambda T}} dx = \frac{e^{-\lambda t} - e^{-\lambda T}}{1 - e^{-\lambda T}}.$$

Obviously as for the upper-bound truncated exponential distribution, the failure rate is not a constant for all $t > T$. To see this, the failure rate of this upper-bound truncated exponential distribution is

$$h(t) = \frac{f(t)}{R(t)} = \frac{\lambda}{1 - e^{-\lambda(T-t)}}.$$

and is a function of t .

Uniform Distribution

Let X be a random variable with a uniform distribution over the interval (a, b) where $a < b$. The pdf is given by

$$f(x) = \begin{cases} \frac{1}{b-a} & a \leq x \leq b \\ 0 & \text{otherwise} \end{cases}.$$

The expected value and variance are, respectively,

$$E(X) = \frac{a+b}{2}$$

and

$$V(X) = \frac{(b-a)^2}{12}.$$

Normal Distribution

The normal distribution plays an important role in classical statistics due to the *central limit theorem*. In production engineering, the normal distribution primarily applies to measurements of product susceptibility and external stress. This two-parameter distribution is used to describe mechanical systems in which a failure results from some wearout effect. The normal distribution takes the well-known bell shape. This distribution is symmetrical about the mean and the spread is measured by the variance. The larger the value, the flatter the distribution. The pdf is given by

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2}, \quad -\infty < t < \infty,$$

where μ is the mean value and σ is the standard deviation. The cumulative distribution function (cdf) is

$$F(t) = \int_{-\infty}^t \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{s-\mu}{\sigma}\right)^2} ds.$$

The reliability function is

$$R(t) = \int_t^{\infty} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{s-\mu}{\sigma}\right)^2} ds.$$

There is no closed-form solution for the above equation. However, tables for the standard normal density function are readily available (see Table 1.8 in Appendix A) and can be used to find probabilities for any normal distribution. If

$$Z = \frac{T - \mu}{\sigma}$$

is substituted into the normal pdf, we obtain

$$f(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}}, \quad -\infty < Z < \infty.$$

This is a so-called standard normal pdf, with a mean value of 0 and a standard deviation of 1. The standardized cdf is given by

$$\Phi(t) = \int_{-\infty}^t \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}s^2} ds, \quad (1.17)$$

where Φ is a standard normal distribution function. Thus, for a normal random variable T , with mean μ and standard deviation σ ,

$$P(T \leq t) = P\left(Z \leq \frac{t - \mu}{\sigma}\right) = \Phi\left(\frac{t - \mu}{\sigma}\right),$$

where Φ yields the relationship required if standard normal tables are to be used.

It should be noted that the coefficient of kurtosis in the normal distribution is 3. The hazard function for a normal distribution is a monotonically increasing function of t . This is easily shown by proving that $h'(t) \geq 0$ for all t . Since

$$h(t) = \frac{f(t)}{R(t)}$$

then

$$h'(t) = \frac{R(t)f'(t) + f^2(t)}{R^2(t)} \geq 0.$$

One can attempt this proof by using the basic definition of a normal density function f .

Example 1.11 A component has a normal distribution of failure times with $\mu = 2000$ h and $\sigma = 100$ h. The reliability of the component at 1900 h is required.

Note that the reliability function is related to the standard normal deviate z by

$$R(t) = P\left(Z > \frac{t - \mu}{\sigma}\right),$$

where the distribution function for Z is given by (1.17). For this particular application,

$$\begin{aligned} R(1900) &= P\left(Z > \frac{1900 - 2000}{100}\right) \\ &= P(z > -1). \end{aligned}$$

From the standard normal table in Table 1.8 in Appendix A, we obtain

$$R(1,900) = 1 - \Phi(-1) = 0.8413.$$

The value of the hazard function is found from the relationship

$$h(t) = \frac{f(t)}{R(t)} = \frac{\Phi\left(\frac{t - \mu}{\sigma}\right)}{\sigma R(t)},$$

where Φ is the pdf of the standard normal density. Here

$$\begin{aligned} h(1900) &= \frac{\Phi(-1.0)}{\sigma R(t)} = \frac{0.1587}{100(0.8413)} \\ &= 0.0019 \text{ failures/cycle.} \end{aligned}$$

The normal distribution is flexible enough to make it a very useful empirical model. It can be theoretical derived under assumptions matching many failure mechanisms. Some of these are: corrosion, migration, crack growth, and failures resulting from chemical reactions or processes in general. That does not mean that the normal distribution is always the correct model for these mechanisms, but it does perhaps explain why it has been empirically successful in so many of these cases.

Log Normal Distribution

The log normal lifetime distribution is a very flexible model that can empirically fit many types of failure data. This distribution, when applied in mechanical reliability engineering, is able to model failure probabilities of repairable systems, the compressive strength of concrete cubes, the tensile strength of fibers, and the uncertainty in failure rate information. The log normal density function is given by

$$f(t) = \frac{1}{\sigma t \sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\ln t - \mu}{\sigma}\right)^2}, \quad t \geq 0, \quad (1.18)$$

where μ and σ are parameters such that $-\infty < \mu < \infty$, and $\sigma > 0$. Note that μ and σ are not the mean and standard deviations of the distribution.

Its relationship to the normal (just take natural logarithms of all of the data and time points and you have “normal” data) makes it easy to work with many good software analysis programs used to treat normal data.

Mathematically, if a random variable X is defined as $X = \ln T$, then X is normally distributed with a mean of μ and a variance of σ^2 . That is,

$$E(X) = E(\ln T) = \mu$$

and

$$V(X) = V(\ln T) = \sigma^2.$$

Since $T = e^X$, the mean of the log normal distribution can be found via the normal distribution. Consider that

$$E(T) = E(e^X) = \int_{-\infty}^{\infty} \frac{1}{\sigma \sqrt{2\pi}} e^{[x - \frac{1}{2}\left(\frac{x - \mu}{\sigma}\right)^2]} dx.$$

By rearranging the exponent, this integral becomes

$$E(T) = e^{\mu + \frac{\sigma^2}{2}} \int_{-\infty}^{\infty} \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2\sigma^2} [x - (\mu + \sigma^2)]^2} dx.$$

Thus, the mean of the log normal distribution is

$$E(T) = e^{\mu + \frac{\sigma^2}{2}}.$$

Proceeding in a similar manner,

$$E(T^2) = E(e^{2X}) = e^{2(\mu + \sigma^2)}$$

so the variance for the log normal is

$$V(T) = e^{2\mu + 2\sigma^2} (e^{\sigma^2} - 1).$$

The coefficient of skewness of this distribution is

$$S_c = \frac{e^{3\sigma^2} - 3e^{\sigma^2} + 2}{(e^{\sigma^2} - 1)^{\frac{3}{2}}}.$$

It is interesting that the skewness coefficient does not depend on μ and grows rapidly as the variance σ^2 increases.

The cumulative distribution function for the log normal is

$$F(t) = \int_0^t \frac{1}{\sigma s \sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{\ln s - \mu}{\sigma} \right)^2} ds$$

and this can be related to the standard normal deviate Z by

$$\begin{aligned} F(t) &= P(T \leq t) = P(\ln T \leq \ln t) \\ &= P\left(Z \leq \frac{\ln t - \mu}{\sigma}\right). \end{aligned}$$

Therefore, the reliability function is given by

$$R(t) = P\left(Z > \frac{\ln t - \mu}{\sigma}\right)$$

and the hazard function would be

$$h(t) = \frac{f(t)}{R(t)} = \frac{\Phi\left(\frac{\ln t - \mu}{\sigma}\right)}{\sigma t R(t)}$$

where Φ is the cdf of standard normal density.

The log normal lifetime model, like the normal, is flexible enough to make it a very useful empirical model. It can be theoretically derived under assumptions matching many failure mechanisms, including corrosion, migration, crack growth, and failures resulting from chemical reactions or processes in general. As with the normal distribution, this

does not mean that the log normal is always the correct model for these mechanisms, but it suggests why it has been empirically successful in so many of these cases.

Student's T Distribution

Student's t probability density function of a random variable T is given by:

$$f(t) = \frac{\Gamma\left(\frac{r+1}{2}\right)}{\sqrt{\pi} \Gamma\left(\frac{r}{2}\right) \left(1 + \frac{t^2}{r}\right)^{\frac{r+1}{2}}} \quad \text{for } -\infty < t < \infty.$$

In other words, if a random variable T is defined as

$$T = \frac{W}{\sqrt{\frac{V}{r}}},$$

where W is a standard normal random variable and V has a chi-square distribution with r degrees of freedom, and W and V are statistically independent, then T is Student's t -distributed, and parameter r is referred to as the degrees of freedom (see Table 1.9 in Appendix A).

The F Distribution

Let us define the random variable F is as follows:

$$F = \frac{U/r_1}{V/r_2},$$

where U has a chi-square distribution with r_1 degrees of freedom, V is also chi-square-distributed, with r_2 degrees of freedom, and U and V are statistically independent, then the probability density function of F is given by

$$f(t) = \frac{\Gamma\left(\frac{r_1+r_2}{2}\right) \left(\frac{r_1}{r_2}\right)^{\frac{r_1}{2}} t^{\frac{r_1}{2}-1}}{\Gamma\left(\frac{r_1}{2}\right) \Gamma\left(\frac{r_2}{2}\right) \left(1 + \frac{r_1 t}{r_2}\right)^{\frac{r_1+r_2}{2}}} \quad \text{for } t > 0.$$

The F distribution is a two-parameter r_1 and r_2 - distribution where the parameters are the degrees of freedom of the underlying chi-square random variables (see Table 1.10 in Appendix A).

It is worth noting that if T is a random variable with a t distribution and r degrees of freedom, then the random variable T^2 has an F distribution with parameters $r_1 = 1$ and $r_2 = r$. Similarly, if F is F -distributed with r_1 and r_2 degrees of freedom, then the random variable Y , defined as

$$Y = \frac{r_1 F}{r_2 + r_1 F}$$

has a beta distribution with parameters $r_1/2$ and $r_2/2$.

Weibull Distribution

The exponential distribution is often limited in applicability owing to its memoryless property. The Weibull distribution [2] is a generalization of the exponential distribution and is commonly used to represent fatigue life, ball-bearing life, and vacuum tube life. The Weibull distribution is extremely flexible and appropriate for modeling component lifetimes with fluctuating hazard rate functions and is used to represent various types of engineering applications. The three-parameter Weibull probability density function is

$$f(t) = \frac{\beta(t - \gamma)^{\beta-1}}{\theta^\beta} e^{-\left(\frac{t-\gamma}{\theta}\right)^\beta}, \quad t \geq \gamma \geq 0, \quad (1.19)$$

where θ and β are known as the scale and shape parameters, respectively, and γ is known as the location parameter. These parameters are always positive. By using different parameters, this distribution can follow the exponential distribution, the normal distribution, etc. It is clear that, for $t \geq \gamma$, the reliability function $R(t)$ is

$$R(t) = e^{-\left(\frac{t-\gamma}{\theta}\right)^\beta} \quad \text{for } t > \gamma > 0, \quad \beta > 0, \theta > 0 \quad (1.20)$$

hence,

$$h(t) = \frac{\beta(t - \gamma)^{\beta-1}}{\theta^\beta}, \quad t > \gamma > 0, \quad \beta > 0, \quad \theta > 0. \quad (1.21)$$

It can be shown that the hazard function decreases for $\beta < 1$, increases for $\beta > 1$, and is constant when $\beta = 1$.

Note that the Rayleigh and exponential distributions are special cases of the Weibull distribution at $\beta = 2$, $\gamma = 0$, and $\beta = 1$, $\gamma = 0$, respectively. For example, when $\beta = 1$ and $\gamma = 0$, the reliability of the Weibull distribution function in (1.20) reduces to

$$R(t) = e^{-\frac{t}{\theta}}$$

and the hazard function given in (1.21) reduces to $1/\theta$, a constant. Thus, the exponential is a special case of the Weibull distribution. Similarly, when $\gamma = 0$ and $\beta = 2$, the Weibull probability density function becomes the Rayleigh density function. That is,

$$f(t) = \frac{2}{\theta} t e^{-\frac{t^2}{\theta}} \quad \text{for } \theta > 0, \quad t \geq 0.$$

Other Forms of Weibull Distributions

The Weibull distribution is widely used in engineering applications. It was originally proposed in order to represent breaking strength distributions of materials. The Weibull model is very flexible and has also been applied in many applications as a purely empirical model, with theoretical justifi-

fication. Other forms of Weibull probability density function include, for example,

$$f(x) = \lambda \gamma x^{\gamma-1} e^{-\lambda x^\gamma}. \quad (1.22)$$

When $\gamma = 2$, the density function becomes a Rayleigh distribution.

It is easy to show that the mean, variance, and reliability of the above Weibull distribution are, respectively:

$$\text{Mean} = \lambda^{\frac{1}{\gamma}} \Gamma\left(1 + \frac{1}{\gamma}\right);$$

$$\text{Variance} = \lambda^{\frac{2}{\gamma}} \left\{ \Gamma\left(1 + \frac{2}{\gamma}\right) - \left[\Gamma\left(1 + \frac{1}{\gamma}\right) \right]^2 \right\};$$

$$\text{Reliability} = e^{-\lambda t^\gamma}.$$

Example 1.12 The time to failure of a part has a Weibull distribution with $1/\lambda = 250$ (measured in 10^5 cycles) and $\gamma = 2$. The part reliability at 10^6 cycles is given by:

$$R(10^6) = e^{-(10^6)^2/250} = 0.6703.$$

The resulting reliability function is shown in Fig. 1.2.

Gamma Distribution

The gamma distribution can be used as a failure probability function for components whose distribution is skewed. The failure density function for a gamma distribution is

$$f(t) = \frac{t^{\alpha-1}}{\beta^\alpha \Gamma(\alpha)} e^{-\frac{t}{\beta}}, \quad t \geq 0, \quad \alpha, \beta > 0, \quad (1.23)$$

where α is the shape parameter and β is the scale parameter. In this expression, $\Gamma(\alpha)$ is the gamma function, which is defined as

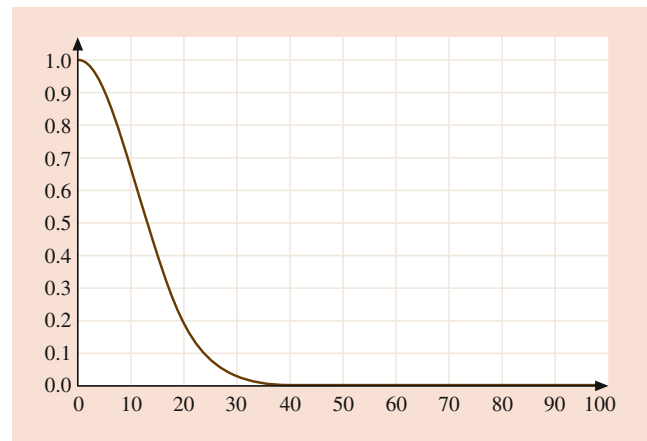


Fig. 1.2 Weibull reliability function versus time

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} dt \quad \text{for } \alpha > 0.$$

Hence,

$$R(t) = \int_t^\infty \frac{1}{\beta^\alpha \Gamma(\alpha)} s^{\alpha-1} e^{-\frac{s}{\beta}} ds.$$

If α is an integer, it can be shown by successive integration by parts that

$$R(t) = e - \frac{t}{\beta} \sum_{i=0}^{\alpha-1} \frac{\left(\frac{t}{\beta}\right)^i}{i!} \quad (1.24)$$

and

$$h(t) = \frac{f(t)}{R(t)} = \frac{\frac{1}{\beta^\alpha \Gamma(\alpha)} t^{\alpha-1} e^{-\frac{t}{\beta}}}{e^{-\frac{t}{\beta}} \sum_{i=0}^{\alpha-1} \frac{\left(\frac{t}{\beta}\right)^i}{i!}}. \quad (1.25)$$

The gamma density function has shapes that are very similar to the Weibull distribution. At $\alpha = 1$, the gamma distribution becomes the exponential distribution with a constant failure rate $1/\beta$. The gamma distribution can also be used to model the time to the n th failure of a system if the underlying failure distribution is exponential. Thus, if X_i is exponentially distributed with parameter $\theta = 1/\beta$, then $T = X_1 + X_2 + \dots + X_n$ is gamma-distributed with parameters β and n .

Example 1.13 The time to failure of a component has a gamma distribution with $\alpha = 3$ and $\beta = 5$. Obtain the reliability of the component and the hazard rate at 10 time units.

Using Eq. (1.24), we compute

$$R(10) = e^{-\frac{10}{5}} \sum_{i=0}^2 \frac{\left(\frac{10}{5}\right)^i}{i!} = 0.6767.$$

The hazard rate is given by

$$h(10) = \frac{f(10)}{R(10)} = \frac{0.054}{0.6767} = 0.798 \text{ failures/unit time.}$$

The other form of the gamma probability density function can be written as follows:

$$f(x) = \frac{\beta^\alpha t^{\alpha-1}}{\Gamma(\alpha)} e^{-t\beta}, \quad t > 0.$$

This pdf is characterized by two parameters: the shape parameter α and the scale parameter β . When $0 < \alpha < 1$, the

failure rate monotonically decreases; when $\alpha > 1$, the failure rate monotonically increases; when $\alpha = 1$ the failure rate is constant.

The mean, variance, and reliability of the gamma random variable are:

$$\text{Mean (MTTF)} = \frac{\alpha}{\beta};$$

$$\text{Variance} = \frac{\alpha}{\beta^2};$$

$$\text{Reliability} = \int_t^\infty \frac{\beta^\alpha x^{\alpha-1}}{\Gamma(\alpha)} e^{-x\beta} dx.$$

Example 1.14 A mechanical system time to failure is gamma-distributed with $\alpha = 3$ and $1/\beta = 120$. The system reliability at 280 h is given by

$$R(280) = e^{-\frac{280}{120}} \sum_{k=0}^2 \frac{\left(\frac{280}{120}\right)^k}{k!} = 0.85119$$

and the resulting reliability plot is shown in Fig. 1.3.

The gamma model is a flexible lifetime model that may offer a good fit to some sets of failure data. Although it is not widely used as a lifetime distribution model for common failure mechanisms, the gamma lifetime model is commonly used in Bayesian reliability applications.

Beta Distribution

The two-parameter beta density function $f(t)$ is given by

$$f(t) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha) \Gamma(\beta)} t^{\alpha-1} (1-t)^{\beta-1}, \quad (1.26)$$

$$0 < t < 1, \alpha > 0, \beta > 0,$$

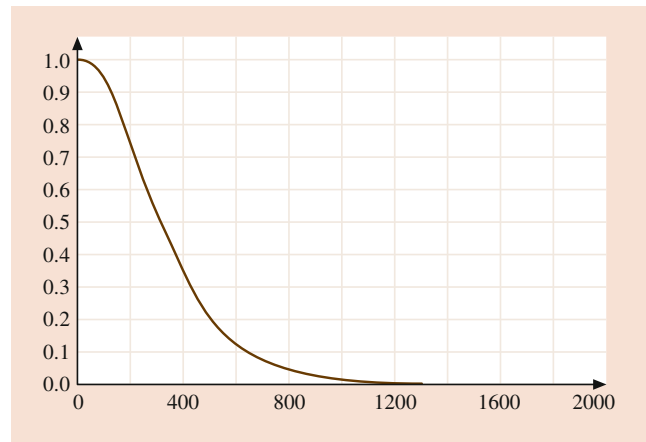


Fig. 1.3 Gamma reliability function versus time

where α and β are the distribution parameters. This two-parameter beta distribution is commonly used in many reliability engineering applications and also plays an important role in the theory of statistics. Note that the beta-distributed random variable takes on values in the interval $(0, 1)$, so the beta distribution is a natural model when the random variable represents a probability. Likewise, when $\alpha = \beta = 1$, the beta distribution reduces to a uniform distribution.

The mean and variance of the beta random variable are, respectively,

$$E(T) = \frac{\alpha}{\alpha + \beta}$$

and

$$V(T) = \frac{\alpha\beta}{(\alpha + \beta + 1)(\alpha + \beta)^2}.$$

Pareto Distribution

The Pareto distribution was originally developed to model income in a population. Phenomena such as city population size, stock price fluctuations, and personal incomes have distributions with very long right tails. The probability density function of the Pareto distribution is given by

$$f(t) = \frac{\alpha k^\alpha}{t^{\alpha+1}}, \quad k \leq t < \infty. \quad (1.27)$$

The mean, variance, and reliability of the Pareto distribution are:

$$\text{Mean} = k/(\alpha - 1) \quad \text{for } \alpha > 1;$$

$$\text{Variance} = \alpha k^2 / [(\alpha - 1)^2 (\alpha - 2)] \quad \text{for } \alpha > 2;$$

$$\text{Reliability} = \left(\frac{k}{t}\right)^\alpha.$$

The Pareto and log normal distributions are commonly used to model population size and economical incomes. The Pareto is used to fit the tail of the distribution, and the log normal is used to fit the rest of the distribution.

Rayleigh Distribution

The Rayleigh model is a flexible lifetime model that can apply to many degradation process failure modes. The Rayleigh probability density function is

$$f(t) = \frac{t}{\sigma^2} \exp\left(-\frac{t^2}{2\sigma^2}\right). \quad (1.28)$$

The mean, variance, and reliability of the Rayleigh function are:

$$\text{Mean} = \sigma \left(\frac{\pi}{2}\right)^{\frac{1}{2}};$$

$$\text{Variance} = \left(2 - \frac{\pi}{2}\right) \sigma^2;$$

$$\text{Reliability} = e^{-\frac{\sigma^2}{2t}}.$$

Example 1.15 Rolling resistance is a measure of the energy lost by a tire under load when it resists the force opposing its direction of travel. In a typical car traveling at 60 miles per hour, about 20% of the engine power is used to overcome the rolling resistance of the tires. A tire manufacturer introduces a new material that, when added to the tire rubber compound, significantly improves the tire rolling resistance but increases the wear rate of the tire tread. Analysis of a laboratory test of 150 tires shows that the failure rate of the new tire increases linearly with time (h). This is expressed as

$$h(t) = 0.5 \times 10^{-8} t.$$

The reliability of the tire after one year (8760 h) of use is

$$R(1 \text{ y}) = e^{-\frac{0.5}{2} \times 10^{-8} \times (8760)^2} = 0.8254.$$

Figure 1.4 shows the resulting reliability function.

Pham Distribution

A two-parameter distribution with a Vtub-shaped hazard rate curve was developed by Pham [3], known as *loglog distribution* or *Pham distribution* [3, 15]. Note that the loglog distribution with a Vtub-shaped hazard rate and the Weibull distribution with bathtub-shaped failure rates are not the same. For the bathtub-shaped failure rate, after an initial “infant mortality period,” the useful life of the system begins. During its useful life, the system fails at a constant rate. This period is then followed by a wearout period during which

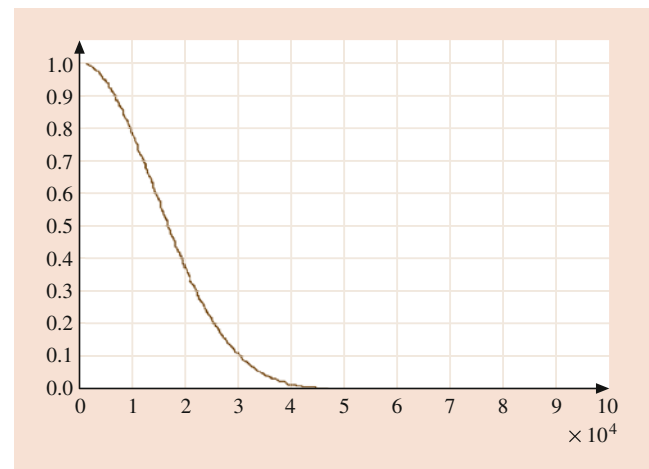


Fig. 1.4 Rayleigh reliability function versus time

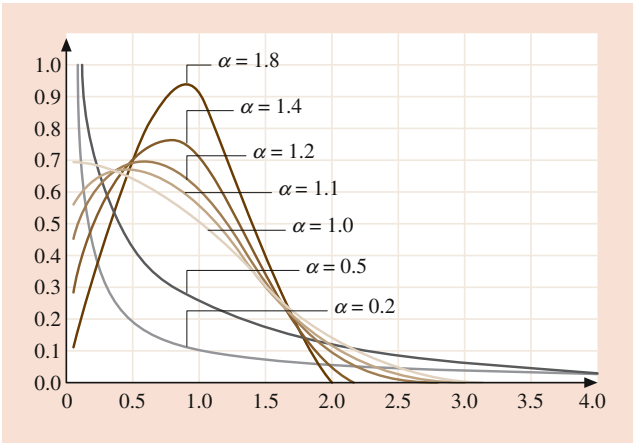


Fig. 1.5 Pham probability density function for various values of α with $a = 2$

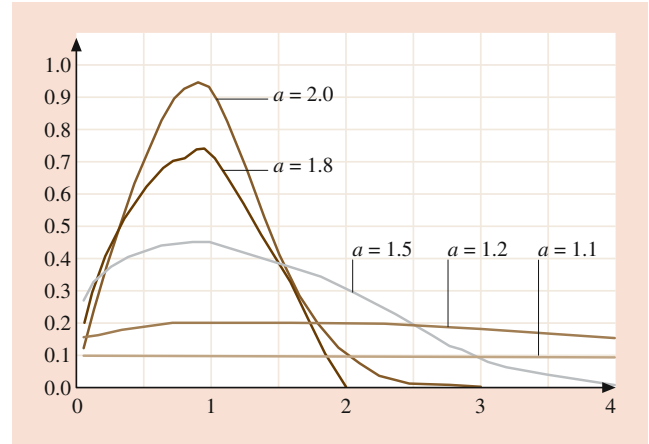


Fig. 1.6 Pham probability density function for various values of a with $\alpha = 1.5$

the system failure rate slowly increases with the onset of wearout. For the Vtub-shaped hazard rate, after the infant mortality period, the system experiences a relatively low but increasing failure rate. The failure rate increases due to aging.

The Pham probability density function is defined as [3]:

$$f(t) = \alpha t^{\alpha-1} a^{t^\alpha} e^{1-a^{t^\alpha}} \ln(a) \quad \text{for } t > 0, a > 1, \alpha > 0. \quad (1.29)$$

The corresponding Pham cdf and reliability function are given by

$$F(t) = \int_0^t f(x)dx = 1 - e^{1-a^{t^\alpha}}$$

and

$$R(t) = e^{1-a^{t^\alpha}}, \quad (1.30)$$

respectively. The corresponding failure rate of the Pham distribution is given by

$$h(t) = \alpha \ln a t^{\alpha-1} a^{t^\alpha}. \quad (1.31)$$

Figures 1.5 and 1.6 describe the density functions and failure rate functions for various values of a and α .

Two-Parameter Hazard Rate Function

This is a two-parameter function that can have increasing and decreasing hazard rates. The hazard rate $h(t)$, the reliability function $R(t)$, and the pdf are, respectively, given as follows:

$$h(t) = \frac{n\lambda t^{n-1}}{\lambda t^n + 1} \quad \text{for } n \geq 1, \lambda > 0, t \geq 0, \quad (1.32)$$

$$R(t) = e^{-\ln(\lambda t^n + 1)}$$

and

$$f(t) = \frac{n\lambda t^{n-1}}{\lambda t^n + 1} e^{-\ln(\lambda t^n + 1)}, \quad n \geq 1, \lambda > 0, t \geq 0,$$

where n = shape parameter and λ = scale parameter.

Three-Parameter Hazard Rate Function

This is a three-parameter distribution that can have increasing and decreasing hazard rates. The hazard rate $h(t)$ is given as

$$h(t) = \frac{\lambda(b+1)[\ln(\lambda t + \alpha)]^b}{(\lambda t + \alpha)}, \quad (1.33)$$

$$b \geq 0, \lambda > 0, \alpha \geq 0, t \geq 0.$$

The reliability function $R(t)$ for $\alpha = 1$ is

$$R(t) = e^{-[\ln(\lambda t + \alpha)]^{b+1}}.$$

The probability density function $f(t)$ is

$$f(t) = e^{-[\ln(\lambda t + \alpha)]^{b+1}} \frac{\lambda(b+1)[\ln(\lambda t + \alpha)]^b}{(\lambda t + \alpha)},$$

where b = shape parameter, λ = scale parameter, and α = location parameter.

Extreme-Value Distribution

The extreme-value distribution can be used to model external events such as floods, tornadoes, hurricanes, and high winds in risk applications. The cdf of this distribution is given by

$$F(t) = e^{-e^y} \quad \text{for } -\infty < t < \infty. \quad (1.34)$$

Cauchy Distribution

The Cauchy distribution can be applied when analyzing communication systems where two signals are received and one is interested in modeling the ratio of the two signals. The Cauchy probability density function is given by

$$f(t) = \frac{1}{\pi(1+t^2)} \quad \text{for } -\infty < t < \infty. \quad (1.35)$$

It is worth noting that the ratio of two standard normal random variables is a random variable with a Cauchy distribution.

Vtub-Shaped Failure Rate

Pham [14] recently presents a distribution function that characterizes a Vtub-shaped failure rate function. Note that, for the bathtub-shaped failure rate, during its useful life period the failure rate is constant. For the Vtub-shaped failure rate, after the infant mortality period the system will begin to experience a rather low increasing failure rate but never be at a constant rate due to aging. The Vtub-shaped failure rate function $h(t)$ is defined as

$$h(t) = at \ln(bt) + \frac{a}{b} \quad \text{for } t > 0, \quad a > 0, \quad b > 0 \quad (1.36)$$

Figure 1.7 shows the Vtub-shaped failure rate $h(t)$ for $a = 0.75$ and various values of b (i.e., 0.35, 0.45, and 0.55). The corresponding probability density function $f(t)$ and reliability function $R(t)$ are as follows:

$$f(t) = \left(at \ln(bt) + \frac{a}{b} \right) e^{-\left\{ at \left[\frac{t}{2} \ln(bt) - \frac{t}{4} + \frac{1}{b} \right] \right\}} \quad (1.37)$$

for $t > 0, a > 0, b > 0$

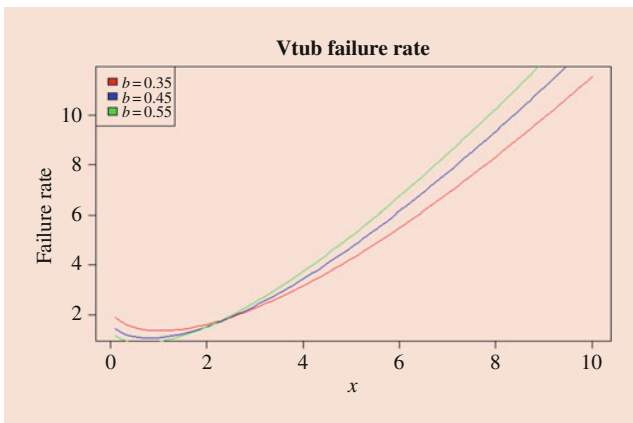


Fig. 1.7 A Vtub-shaped failure rate for $a = 0.75$ and various values of b (i.e. 0.35, 0.45, and 0.55)

and

$$R(t) = e^{-at \left[\frac{t}{2} \ln(bt) - \frac{t}{4} + \frac{1}{b} \right]} \quad \text{for } t > 0, \quad a > 0, \quad b > 0 \quad (1.38)$$

respectively.

Triangular Distribution

A triangular distribution (sometime called a triangle distribution) is a continuous probability distribution shaped like a triangle. It is defined by

a : the minimum value

c : the peak value (the height of the triangle)

b : the maximum value, where $a \leq c \leq b$.

The triangular probability density function is given by

$$f(t) = \begin{cases} 0 & t < a \\ \frac{2(t-a)}{(b-a)(c-a)} & a \leq t \leq c \\ \frac{2(b-t)}{(b-a)(b-c)} & c \leq t \leq b \\ 0 & t > b \end{cases} \quad (1.39)$$

The three parameters a , b , and c change the shape of the triangle. The mean and the variance for this distribution, respectively, are:

$$\text{mean} = \frac{1}{3}(a + b + c)$$

and

$$\text{variance} = \frac{1}{18}(a^2 + b^2 + c^2 - ab - ac - bc).$$

1.4 Statistical Inference and Estimation

Inference about the values of parameters involved in statistical distributions is known as estimation. The engineer might be interested in estimating the mean life of an electronic device based on the failure times of a random sample placed on life test. An interesting question that commonly is asked from many practitioners is how close this estimator would be to the true value of the parameter being estimated from a known distribution.

The problem of “point estimation” is that of estimating the parameters of a population, such as λ or θ from an exponential, μ and σ^2 from a normal, etc. It is assumed that the type of population distribution involved is known, but the distribution parameters are unknown and they must be estimated using collected failure data. This section is devoted to the theory of estimation and discusses several common estimation techniques, such as maximum likelihood, method

of moments, least squared estimation, and Bayesian methods. We also discuss confidence interval and tolerance limit estimation. For example, assume that n independent samples are drawn from the exponential density function $f(x; \lambda) = \lambda e^{-\lambda x}$ for $x > 0$ and $\lambda > 0$. Then the joint probability density function (pdf) or sample density (for short) is given by

$$f(x_1, \lambda) \cdot f(x_2, \lambda) \cdots f(x_n, \lambda) = \lambda^n e^{-\lambda \sum_{i=1}^n x_i}. \quad (1.40)$$

The problem here is to find a “good” point estimate of λ , which is denoted by $\hat{\lambda}$. In other words, we want to find a function $h(X_1, X_2, \dots, X_n)$ such that, if x_1, x_2, \dots, x_n are the observed experimental values of X_1, X_2, \dots, X_n , the value $h(x_1, x_2, \dots, x_n)$ will be a good point estimate of λ . By “good,” we mean that it possesses the following properties:

- unbiasedness
- consistency
- efficiency (minimum variance)
- sufficiency

In other words, if $\hat{\lambda}$ is a good point estimate of λ , then one can select a function $h(X_1, X_2, \dots, X_n)$ where $h(X_1, X_2, \dots, X_n)$ is an unbiased estimator of λ and the variance of $h(X_1, X_2, \dots, X_n)$ is a minimum. We will now present the following definitions.

Unbiasedness

For a given positive integer n , the statistic $Y = h(X_1, X_2, \dots, X_n)$ is called an unbiased estimator of the parameter θ if the expectation of Y is equal to a parameter θ ; that is,

$$E(Y) = \theta.$$

Consistency

The statistic Y is called a consistent estimator of the parameter θ if Y converges stochastically to a parameter θ as n approaches infinity. If ε is an arbitrarily small positive number when Y is consistent, then

$$\lim_{n \rightarrow \infty} P(|Y - \theta| \leq \varepsilon) = 1.$$

Minimum Variance

The statistic Y is called the minimum variance unbiased estimator of the parameter θ if Y is unbiased and the variance of Y is less than or equal to the variance of every other unbiased estimator of θ . An estimator that has the property of minimum variance in large samples is said to be efficient.

Sufficiency

The statistic Y is said to be sufficient for θ if the conditional distribution of X , given that $Y = y$, is independent of θ .

This is useful when determining a lower bound on the variance of all unbiased estimators. We now establish a lower-bound inequality known as the *Cramér–Rao inequality*.

Cramér–Rao Inequality

Let X_1, X_2, \dots, X_n denote a random sample from a distribution with pdf $f(x; \theta)$ for $\theta_1 < \theta < \theta_2$, where θ_1 and θ_2 are known. Let $Y = h(X_1, X_2, \dots, X_n)$ be an unbiased estimator of θ . The lower-bound inequality on the variance of Y , $\text{Var}(Y)$, is given by

$$\text{Var}(Y) \geq \frac{1}{nE \left[\left(\frac{\partial \ln f(x; \theta)}{\partial \theta} \right)^2 \right]} = \frac{1}{-nE \left(\frac{\partial^2 \ln f(x; \theta)}{\partial \theta^2} \right)}. \quad (1.41)$$

Theorem 1.5 An estimator $\hat{\theta}$ is said to be asymptotically efficient if $\sqrt{n}\hat{\theta}$ has a variance that approaches the Cramér–Rao lower bound for large n ; that is,

$$\lim_{n \rightarrow \infty} \text{Var}(\sqrt{n}\hat{\theta}) = \frac{1}{-nE \left(\frac{\partial^2 \ln f(x; \theta)}{\partial \theta^2} \right)}. \quad (1.42)$$

1.4.1 Parameter Estimation

We now discuss some basic methods of parameter estimation, including the method of maximum likelihood estimation (MLE) and the method of moments. The assumption that the sample is representative of the population will be made both in the example and in later discussions.

Maximum Likelihood Estimation Method

In general, one deals with a sample density

$$f(x_1, x_2, \dots, x_n) = f(x_1; \theta) f(x_2; \theta) \cdots f(x_n; \theta),$$

where x_1, x_2, \dots, x_n are random, independent observations of a population with density function $f(x)$.

For the general case, we would like to find an estimate or estimates, $\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_m$ (if such exist), where

$$\begin{aligned} f(x_1, x_2, \dots, x_n; \theta_1, \theta_2, \dots, \theta_m) > \\ f(x_1, x_2, \dots, x_n; \theta'_1, \theta'_2, \dots, \theta'_m). \end{aligned}$$

The notation $\theta'_1, \theta'_2, \dots, \theta'_m$ refers to any other estimates different to $\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_m$.

Consider a random sample X_1, X_2, \dots, X_n from a distribution with a pdf $f(x; \theta)$. This distribution has a vector $\theta = (\theta_1, \theta_2, \dots, \theta_m)'$ of unknown parameters associated with it, where m is the number of unknown parameters. Assuming that the random variables are independent, then the likelihood function, $L(X; \theta)$, is the product of the probability density function evaluated at each sample point

$$L(X, \theta) = \prod_{i=1}^n f(X_i; \theta), \quad (1.43)$$

where $\mathbf{X} = (X_1, X_2, \dots, X_n)$. The maximum likelihood estimator $\hat{\theta}$ is found by maximizing $L(\mathbf{X}; \theta)$ with respect to θ . In practice, it is often easier to maximize $\ln [L(\mathbf{X}; \theta)]$ in order to find the vector of MLEs, which is valid because the logarithmic function is monotonic. The log likelihood function is given by

$$\ln L(X, \theta) = \prod_{i=1}^n \ln f(X_i; \theta) \quad (1.44)$$

and is asymptotically normally distributed since it consists of the sum of n independent variables and the central limit theorem is implied. Since $L(X; \theta)$ is a joint probability density function for X_1, X_2, \dots, X_n , its integral must be 1; that is,

$$\int_0^\infty \int_0^\infty \cdots \int_0^\infty L(X; \theta) dX = 1.$$

Assuming that the likelihood is continuous, the partial derivative of the left-hand side with respect to one of the parameters, θ_i , yields

$$\begin{aligned} & \frac{\partial}{\partial \theta_i} \int_0^\infty \int_0^\infty \cdots \int_0^\infty L(X; \theta) dX \\ &= \int_0^\infty \int_0^\infty \cdots \int_0^\infty \frac{\partial}{\partial \theta_i} L(X; \theta) dX \\ &= \int_0^\infty \int_0^\infty \cdots \int_0^\infty \frac{\partial \log L(X; \theta)}{\partial \theta_i} L(X; \theta) dX \\ &= E \left(\frac{\partial \log L(X; \theta)}{\partial \theta_i} \right) \\ &= E [U_i(\theta)] \quad \text{for } i = 1, 2, \dots, m, \end{aligned}$$

where $\mathbf{U}(\theta) = [U_1(\theta), U_2(\theta), \dots, U_m(\theta)]'$ is often called the score vector, and the vector $\mathbf{U}(\theta)$ has components

$$U_i(\theta) = \frac{\partial [\log L(X; \theta)]}{\partial \theta_i} \quad \text{for } i = 1, 2, \dots, m \quad (1.45)$$

which, when equated to zero and solved, yields the MLE vector $\hat{\theta}$.

Suppose that we can obtain a nontrivial function of X_1, X_2, \dots, X_n , say $h(X_1, X_2, \dots, X_n)$, such that, when θ is replaced by $h(X_1, X_2, \dots, X_n)$, the likelihood function L will achieve a maximum. In other words,

$$L[X, h(X)] \geq L(X, \theta)$$

for every θ . The statistic $h(X_1, X_2, \dots, X_n)$ is called a maximum likelihood estimator of θ and will be denoted as

$$\hat{\theta} = h(x_1, x_2, \dots, x_n). \quad (1.46)$$

The observed value of $\hat{\theta}$ is called the MLE of θ . In general, the mechanics for obtaining the MLE can be obtained as follows:

Step 1.	Find the joint density function $L(X, \theta)$
Step 2.	Take the natural log of the density $\ln L$
Step 3.	Find the partial derivatives of $\ln L$ with respect to each parameter
Step 4.	Set these partial derivatives to “zero”
Step 5.	Solve for parameter(s)

Example 1.16 Let X_1, X_2, \dots, X_n , denote a random sample from the normal distribution $N(\mu, \sigma^2)$. Then the likelihood function is given by

$$L(X, \mu, \sigma^2) = \left(\frac{1}{2\pi} \right)^{\frac{n}{2}} \frac{1}{\sigma^n} e^{-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2}$$

and

$$\ln L = -\frac{n}{2} \log(2\pi) - \frac{n}{2} \log \sigma^2 - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2.$$

Thus, we have

$$\begin{aligned} \frac{\partial \ln L}{\partial \mu} &= \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu) = 0, \\ \frac{\partial \ln L}{\partial \sigma^2} &= -\frac{n}{2\sigma^2} - \frac{1}{2\sigma^4} \sum_{i=1}^n (x_i - \mu)^2 = 0. \end{aligned}$$

Solving the two equations simultaneously, we obtain

$$\begin{aligned} \hat{\mu} &= \frac{\sum_{i=1}^n x_i}{n}, \\ \hat{\sigma}^2 &= \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2. \end{aligned}$$

Note that the MLEs, if they exist, are both sufficient and efficient estimates. They also have an additional property called invariance – in other words, for an MLE of θ , $\mu(\theta)$ is the MLE of $\mu(\theta)$. However, they are not necessarily unbiased (i.e., $E(\hat{\theta}) = \theta$). In fact, the point is that

$$E(\hat{\sigma}^2) = \left(\frac{n-1}{n} \right) \sigma^2 \neq \sigma^2.$$

Therefore, for small n , σ^2 is usually adjusted to account for this bias, and the best estimate of σ^2 is

$$\hat{\sigma}^2 = \left(\frac{1}{n-1} \right) \sum_{i=1}^n (x_i - \bar{x})^2.$$

Sometimes it is difficult to directly obtain maximum likelihood estimators by taking the partial derivatives of the log of likelihood function as mentioned in Step 3 but need to consider another appropriate step, which now leads to a given example below.

Example 1.17 Let X_1, X_2, \dots, X_n represent a random sample from the distribution with pdf

$$f(x; \theta) = e^{-(x-\theta)} \quad \text{for } \theta \leq x \leq \infty \text{ and } -\infty < \theta < \infty$$

The likelihood function is given by

$$\begin{aligned} L(\theta; X) &= \prod_{i=1}^n f(x_i; \theta) \quad \text{for } \theta \leq x_i \leq \infty \text{ all } i \\ &= \prod_{i=1}^n e^{-(x_i-\theta)} = e^{-\sum_{i=1}^n x_i + n\theta} \end{aligned}$$

For fixed values of x_1, x_2, \dots, x_n , we wish to find the value of θ which maximizes $L(\theta; X)$. Here we cannot use the techniques of calculus to maximize $L(\theta; X)$. Note that $L(\theta; X)$ is largest when θ is as large as possible. However, the largest value of θ is equal to the smallest value of X_i in the sample. Thus,

$$\hat{\theta} = \min \{X_i\} \quad 1 \leq i \leq n.$$

Sometimes it is difficult, if not impossible, to obtain maximum likelihood estimators in a closed form, and therefore numerical methods must be used to maximize the likelihood function as we now discuss the following example.

Example 1.18 Suppose that X_1, X_2, \dots, X_n is a random sample from the Weibull distribution with pdf

$$f(x, \alpha, \lambda) = \alpha \lambda x^{\alpha-1} e^{-\lambda x^\alpha}.$$

The likelihood function is

$$L(X, \alpha, \lambda) = \alpha^n \lambda^n \prod_{i=1}^n x_i^{\alpha-1} e^{-\lambda \sum_{i=1}^n x_i^\alpha}.$$

Then

$$\ln L = n \log \alpha + n \log \lambda + (\alpha - 1) \sum_{i=1}^n \log x_i - \lambda \sum_{i=1}^n x_i^\alpha,$$

$$\frac{\partial \ln L}{\partial \alpha} = \frac{n}{\alpha} + \sum_{i=1}^n \log x_i - \lambda \sum_{i=1}^n x_i^\alpha \log x_i = 0,$$

$$\frac{\partial \ln L}{\partial \lambda} = \frac{n}{\lambda} - \sum_{i=1}^n x_i^\alpha = 0.$$

As noted, solutions of the above two equations for α and λ are extremely difficult to obtain and require the application of either graphical or numerical methods. It is sometimes desirable to use a quick method of estimation, which leads to a discussion of the method of moments.

Method of Moments

Here one simply sets the sample moments equal to the corresponding population moments. For example, for the gamma distribution, the mean and the variance of the distribution are, respectively, $\frac{\alpha}{\beta}$ and $\frac{\alpha}{\beta^2}$. Therefore, one has the following two equations in two unknowns:

$$\begin{aligned} \bar{X} &= \frac{\alpha}{\beta}, \\ S^2 &= \frac{\alpha}{\beta^2}. \end{aligned}$$

Solving these two equations simultaneously, we obtain

$$\begin{aligned} \alpha &= \frac{\bar{X}^2}{S^2}, \\ \beta &= \frac{\bar{X}}{S^2}. \end{aligned}$$

1.4.2 Maximum Likelihood Estimation with Censored Data

Censored data arises when we monitor for a random variable of interest – unit failure, for example – but the monitoring is stopped before measurements are complete (i.e., before the unit fails). In other words, censored observation contains only partial information about the random variable of interest. In this section, we consider two types of censoring. The first type of censoring is called Type I censoring, where the event is only observed if it occurs prior to some prespecified time. The second type of censoring is Type II censoring, in which the study continues until the failure of the first r units (or components), where r is some predetermined integer ($r < n$).

Examples of Type II censoring are often used when testing equipment life. Here our items are tested at the same time, and the test is terminated when r of the n items have failed. These approaches may save time and resources because it may take a very long time for all of the items to fail. Both Type I and Type II censoring arise in many reliability applications.

For example, let us say that we have a batch of transistors or tubes. We begin to test them all at $t = 0$, and record their times to failure. Some transistors may take a long time to burn out, and we will not want to wait that long to end the experiment. We might stop the experiment at a prespecified

time t_c , in which case we have Type I censoring. On the other hand, we might not know what fixed value to use for the censoring time beforehand, so we decide to wait until a prespecified number of units have failed, r , in which case we have Type II censoring.

Censoring times may vary from individual to individual or from application to application. We now discuss a general case known as multiple censored data.

Parameter Estimate with Multiple Censored Data

The likelihood function for multiple censored data is given by

$$\begin{aligned} L &= f(t_{1,f}, \dots, t_{r,f}, t_{1,s}, \dots, t_{m,s}) \\ &= C \prod_{i=1}^r f(t_{i,f}) \prod_{j=1}^m [1 - F(t_{j,s})], \end{aligned} \quad (1.47)$$

where C is a constant, $f(\cdot)$ is the density function, and $F(\cdot)$ is the distribution function. There are r failures at times $t_{1,f}, \dots, t_{r,f}$ and m units with censoring times $t_{1,s}, \dots, t_{m,s}$.

Note that we obtain Type I censoring by simply setting $t_{i,f} = t_{i,n}$ and $t_{j,s} = t_0$ in the likelihood function in (1.47). The likelihood function for Type II censoring is similar to Type I censoring except $t_{j,s} = t_r$ in (1.47). In other words, the likelihood function for the first r observations from a sample of size n drawn from the model in both Type I and Type II censoring is given by

$$L = f(t_{1,n}, \dots, t_{r,n}) = C \prod_{i=1}^r f(t_{i,n}) [1 - F(t_*)]^{n-r}, \quad (1.48)$$

where $t_* = t_0$, the time of cessation of the test for Type I censoring, and $t_* = t_r$, the time of the r th failure for Type II censoring.

Example 1.19 Consider a two-parameter probability density distribution with multiple censored data and a distribution function with bathtub-shaped failure rate, as given by [4]:

$$f(t) = \lambda \beta t^{\beta-1} \exp \left[t^\beta + \lambda (1 - e^{t^\beta}) \right], \quad t, \lambda, \beta > 0 \quad (1.49)$$

and

$$F(t) = 1 - \exp \left[\lambda (1 - e^{t^\beta}) \right], \quad t, \lambda, \beta > 0, \quad (1.50)$$

respectively.

Substituting the functions $f(t)$ and $F(t)$ into (1.49) and (1.50) into (1.48), we obtain the logarithm of the likelihood function:

$$\begin{aligned} \ln L &= \ln C + r \ln \lambda + r \ln \beta + \sum_{i=1}^r (\beta - 1) \ln t_i \\ &+ (m + r) \lambda + \sum_{i=1}^r t_i^\beta - \left[\sum_{i=1}^r \lambda e^{t_i^\beta} + \sum_{j=1}^m \lambda e^{t_j^\beta} \right]. \end{aligned}$$

The function $\ln L$ can be maximized by setting the partial derivative of $\ln L$ with respect to λ and β equal to zero, and solving the resulting equations simultaneously for λ and β . Therefore, we obtain

$$\frac{\partial \ln L}{\partial \lambda} = \frac{r}{\lambda} + (m + r) - \sum_{i=1}^r e^{t_i^\beta} - \sum_{j=1}^m e^{t_j^\beta} \equiv 0,$$

$$\begin{aligned} \frac{\partial \ln L}{\partial \beta} &= \frac{r}{\beta} + \sum_{i=1}^r \ln t_i + \sum_{i=1}^r t_i^\beta \ln t_i \\ &- \lambda \left(\sum_{i=1}^r e^{t_i^\beta} t_i^\beta \ln t_i + \sum_{j=1}^m e^{t_j^\beta} t_j^\beta \ln t_j \right) \equiv 0. \end{aligned}$$

This implies that

$$\hat{\lambda} = \frac{r}{\left(\sum_{i=1}^r e^{t_i^{\hat{\beta}}} + \sum_{j=1}^m e^{t_j^{\hat{\beta}}} \right) - m - r} \quad (1.51)$$

and that $\hat{\beta}$ is the solution of

$$\begin{aligned} \frac{r}{\hat{\beta}} + \sum_{i=1}^r \ln t_i + \sum_{i=1}^r t_i^{\hat{\beta}} \ln t_i \\ &= \frac{r}{\left(\sum_{i=1}^r e^{t_i^{\hat{\beta}}} + \sum_{j=1}^m e^{t_j^{\hat{\beta}}} \right) - m - r} \\ &\left(\sum_{i=1}^r e^{t_i^{\hat{\beta}}} t_i^{\hat{\beta}} \ln t_i + \sum_{j=1}^m e^{t_j^{\hat{\beta}}} t_j^{\hat{\beta}} \ln t_j \right). \end{aligned} \quad (1.52)$$

We now discuss two special cases.

Case 1: Type I or Type II Censored Data

From Eq. (1.48), the likelihood function for the first r observations from a sample of size n drawn from the model in both Type I and Type II censoring is

$$L = f(t_{1,n}, \dots, t_{r,n}) = C \prod_{i=1}^r f(t_{i,n}) [1 - F(t_*)]^{n-r},$$

where $t_* = t_0$, the test cessation time for Type I censoring, and $t_* = t_r$, the time of the r th failure for Type II censoring. Equations (1.51) and (1.52) become

$$\begin{aligned}\hat{\lambda} &= \frac{r}{\sum_{i=1}^r e^{t_i^{\hat{\beta}}} + (n-r)e^{t_*^{\hat{\beta}}} - n}, \\ \frac{r}{\hat{\beta}} + \sum_{i=1}^r \ln t_i + \sum_{i=1}^r t_i^{\hat{\beta}} \ln t_i \\ &= \frac{r}{\sum_{i=1}^r e^{t_i^{\hat{\beta}}} + (n-r)e^{t_*^{\hat{\beta}}} - n} \\ &\quad \times \left(\sum_{i=1}^r e^{t_i^{\hat{\beta}}} t_i^{\hat{\beta}} \ln t_i + \sum_{j=1}^m e^{t_j^{\hat{\beta}}} t_j^{\hat{\beta}} \ln t_j \right)\end{aligned}$$

Case 2: Complete Censored Data

Simply replace r with n in (1.51) and (1.52) and ignore the t_j portions. The maximum likelihood equations for the λ and β are given by

$$\begin{aligned}\hat{\lambda} &= \frac{n}{\sum_{i=1}^n e^{t_i^{\hat{\beta}}} - n}, \\ \frac{n}{\hat{\beta}} + \sum_{i=1}^n \ln t_i + \sum_{i=1}^n t_i^{\hat{\beta}} \ln t_i \\ &= \frac{n}{\sum_{i=1}^n e^{t_i^{\hat{\beta}}} - n} \times \sum_{i=1}^n e^{t_i^{\hat{\beta}}} t_i^{\hat{\beta}} \ln t_i.\end{aligned}$$

Confidence Intervals of Estimates

The asymptotic variance–covariance matrix for the parameters (λ and β) is obtained by inverting the Fisher information matrix

$$I_{ij} = E \left(-\frac{\partial^2 L}{\partial \theta_i \partial \theta_j} \right), \quad i, j = 1, 2, \quad (1.53)$$

where $\theta_1, \theta_2 = \lambda$ or β [5]. This leads to

$$\begin{aligned}&\begin{pmatrix} \text{Var}(\hat{\lambda}) & \text{Cov}(\hat{\lambda}, \hat{\beta}) \\ \text{Cov}(\hat{\lambda}, \hat{\beta}) & \text{Var}(\hat{\beta}) \end{pmatrix} \\ &= \begin{pmatrix} E \left(-\frac{\partial^2 \ln L}{\partial^2 \lambda} \Big|_{\hat{\lambda}, \hat{\beta}} \right) & E \left(-\frac{\partial^2 \ln L}{\partial \lambda \partial \beta} \Big|_{\hat{\lambda}, \hat{\beta}} \right) \\ E \left(-\frac{\partial^2 \ln L}{\partial \beta \partial \lambda} \Big|_{\hat{\lambda}, \hat{\beta}} \right) & E \left(-\frac{\partial^2 \ln L}{\partial^2 \beta} \Big|_{\hat{\lambda}, \hat{\beta}} \right) \end{pmatrix}. \quad (1.54)\end{aligned}$$

We can obtain approximate $(1 - \alpha)100\%$ confidence intervals for the parameters λ and β based on the asymptotic normality of the MLEs [5] as:

$$\hat{\lambda} \pm Z_{\alpha/2} \sqrt{\text{Var}(\hat{\lambda})} \quad \text{and} \quad \hat{\beta} \pm Z_{\alpha/2} \sqrt{\text{Var}(\hat{\beta})}, \quad (1.55)$$

where $Z_{\alpha/2}$ is the upper percentile of the standard normal distribution.

Application 1

Consider the lifetime of a part from a helicopter's main rotor blade. Data on lifetime of the part taken a system database collected from October 1995 to September 1999 [3] are shown in Table 1.2. In this application, we consider several distribution functions for this data, including Weibull, log normal, normal, and loglog distribution functions.

The Pham pdf with parameters a and α is

$$f(t) = \alpha (\ln a) t^{\alpha-1} a^\alpha e^{1-a^\alpha} \quad \text{for } t > 0, \alpha > 0, a > 1$$

and its corresponding log likelihood function (1.44) is

$$\begin{aligned}\log L(a, \alpha) &= n \log \alpha + n \ln(\ln a) \\ &\quad + (\alpha - 1) \left(\sum_{i=1}^n \ln t_i \right) \\ &\quad + \ln a \cdot \sum_{i=1}^n t_i^\alpha + n - \sum_{i=1}^n a^{t_i^\alpha}.\end{aligned}$$

We then determine the confidence intervals for parameter estimates a and α . From the above log likelihood function, we can obtain the Fisher information matrix \mathbf{H} as $\mathbf{H} = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix}$, where

$$h_{11} = E \left(-\frac{\partial^2 \log L}{\partial a^2} \right),$$

$$h_{12} = h_{21} = E \left(-\frac{\partial^2 \log L}{\partial a \partial \alpha} \right),$$

$$h_{22} = E \left(-\frac{\partial^2 \log L}{\partial \alpha^2} \right),$$

The variance matrix \mathbf{V} can be obtained as follows:

$$\mathbf{V} = (\mathbf{H})^{-1} \begin{pmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{pmatrix}. \quad (1.56)$$

Table 1.2 Main rotor blade data

Part code	Time to failure (h)
xxx-015-001-107	1634.3
xxx-015-001-107	1100.5
xxx-015-001-107	1100.5
xxx-015-001-107	819.9
xxx-015-001-105	1398.3
xxx-015-001-107	1181
xxx-015-001-107	128.7
xxx-015-001-107	1193.6
xxx-015-001-107	254.1
xxx-015-001-107	3078.5
xxx-015-001-107	3078.5
xxx-015-001-107	3078.5
xxx-015-001-107	26.5
xxx-015-001-107	26.5
xxx-015-001-107	3265.9
xxx-015-001-107	254.1
xxx-015-001-107	2888.3
xxx-015-001-107	2080.2
xxx-015-001-107	2094.3
xxx-015-001-107	2166.2
xxx-015-001-107	2956.2
xxx-015-001-107	795.5
xxx-015-001-107	795.5
xxx-015-001-107	204.5
xxx-015-001-107	204.5
xxx-015-001-107	1723.2
xxx-015-001-107	403.2
xxx-015-001-107	2898.5
xxx-015-001-107	2869.1
xxx-015-001-107	26.5
xxx-015-001-107	26.5
xxx-015-001-107	3180.6
xxx-015-001-107	644.1
xxx-015-001-107	1898.5
xxx-015-001-107	3318.2
xxx-015-001-107	1940.1
xxx-015-001-107	3318.2
xxx-015-001-107	2317.3
xxx-015-001-107	1081.3
xxx-015-001-107	1953.5
xxx-015-001-107	2418.5
xxx-015-001-107	1485.1
xxx-015-001-107	2663.7
xxx-015-001-107	1778.3
xxx-015-001-107	1778.3
xxx-015-001-107	2943.6
xxx-015-001-107	2260
xxx-015-001-107	2299.2
xxx-015-001-107	1655
xxx-015-001-107	1683.1
xxx-015-001-107	1683.1
xxx-015-001-107	2751.4

The variances of a and α are

$$\text{Var}(a) = v_{11} \quad \text{Var}(\alpha) = v_{22}.$$

One can approximately obtain the $(1 - \beta)100\%$ confidence intervals for a and α based on the normal distribution as $\left[\hat{a} - z_{\frac{\beta}{2}}\sqrt{v_{11}}, \hat{a} + z_{\frac{\beta}{2}}\sqrt{v_{11}}\right]$ and $\left[\hat{\alpha} - z_{\frac{\beta}{2}}\sqrt{v_{22}}, \hat{\alpha} + z_{\frac{\beta}{2}}\sqrt{v_{22}}\right]$, respectively, where v_{ij} is given in (1.56) and z_{β} is $\left(1 - \frac{\beta}{2}\right)100\%$ of the standard normal distribution. Having obtained \hat{a} and $\hat{\alpha}$, the MLE of the reliability function can be computed as:

$$\hat{R}(t) = e^{1-\hat{a}t^{\hat{\alpha}}}. \quad (1.57)$$

Let us define a partial derivative vector for reliability $R(t)$ as:

$$v[R(t)] = \left(\frac{\partial R(t)}{\partial a} \quad \frac{\partial R(t)}{\partial \alpha} \right)$$

Then the variance of $R(t)$ can be obtained as:

$$\text{Var}[R(t)] = v[R(t)]V(v[R(t)])^T,$$

where V is given in (1.56).

One can approximately obtain the $(1 - \beta)100\%$ confidence interval for $R(t)$ as

$$\left[\hat{R}(t) - z_{\beta}\sqrt{\text{Var}[R(t)]}, \hat{R}(t) + z_{\beta}\sqrt{\text{Var}[R(t)]} \right].$$

The MLE parameter estimations for the loglog distribution and its corresponding parameters, based on the dataset shown in Table 1.2, are:

$$\begin{aligned} \hat{\alpha} &= 1.1075, \quad \text{Var}(\hat{\alpha}) = 0.0162, \\ 95\% \text{CI for } \hat{\alpha} &: [0.8577, 1.3573]; \\ \hat{a} &= 1.0002, \quad \text{Var}(\hat{a}) = 2.782e^{-08}, \\ 95\% \text{CI for } a &: [0.9998, 1.0005]. \end{aligned}$$

Similarly, the C.I. for $R(t)$ can be obtained directly using (1.53).

1.4.3 Statistical Change-Point Estimation Methods

The change-point problem has been widely studied in reliability applications in areas such as biological sciences, survival analysis, and environmental statistics.

Assume that there is a sequence of random variables X_1, X_2, \dots, X_n , that represent the inter-failure times, and that an index change-point τ exists, such that $X_1, X_2, \dots, X_{\tau}$

have a common distribution F with a density function $f(t)$ and $X_{\tau+1}, X_{\tau+2}, \dots, X_n$ have a distribution G with a density function $g(t)$, where $F \neq G$. Consider the following assumptions:

1. There is a finite but unknown number of units N to be tested.
2. At the beginning, all of the units have the same lifetime distribution F . After τ failures are observed, the remaining $(N - \tau)$ items have the distribution G . The change-point τ is assumed unknown.
3. The sequence $\{X_1, X_2, \dots, X_\tau\}$ is statistically independent of the sequence $\{X_{\tau+1}, X_{\tau+2}, \dots, X_n\}$.
4. The lifetime test is performed according to the Type II censoring approach, in which the number of failures n is predetermined.

Note that the total number of units to put up for testing N can be determined in advance in hardware reliability testing. However, in software reliability testing, the parameter N can be defined as the initial number of faults in the software, and it can be considered to be an unknown parameter. Let T_1, T_2, \dots, T_n be the arrival times for sequential failures. Then

$$\begin{aligned} T_1 &= X_1, \\ T_2 &= X_1 + X_2, \\ &\vdots \\ T_n &= X_1 + X_2 + \dots + X_n. \end{aligned} \quad (1.58)$$

The failure times T_1, T_2, \dots, T_τ are the first τ order statistics of a sample of size N from the distribution F . The failure times $T_{\tau+1}, T_{\tau+2}, \dots, T_n$ are the first $(n - \tau)$ order statistics of a sample of size $(N - \tau)$ from the distribution G .

Example 1.20 The Weibull change-point model of the lifetime distributions F and G with parameters (λ_1, β_1) and (λ_2, β_2) , respectively, can be expressed as:

$$F(t) = 1 - \exp(-\lambda_1 t^{\beta_1}), \quad (1.59)$$

$$G(t) = 1 - \exp(-\lambda_2 t^{\beta_2}). \quad (1.60)$$

Assume that the distributions belong to parametric families $\{F(t|\theta_1), \theta_1 \in \Theta_1\}$ and $\{G(t|\theta_2), \theta_2 \in \Theta_2\}$. Assume that T_1, T_2, \dots, T_τ are the first τ order statistics of a sample of size N from the distribution $\{F(t|\theta_1), \theta_1 \in \Theta_1\}$ and that $T_{\tau+1}, T_{\tau+2}, \dots, T_n$ are the first $(n - \tau)$ order statistics of a sample of size $(N - \tau)$ from the distribution $\{G(t|\theta_2), \theta_2 \in \Theta_2\}$, where N is unknown. The log likelihood function can be expressed as follows [6]:

$$\begin{aligned} L(\tau, N, \theta_1, \theta_2 | T_1, T_2, \dots, T_n) &= \sum_{i=1}^{\tau} (N - i + 1) + \sum_{i=1}^{\tau} f(T_i | \theta_1) \\ &+ \sum_{i=\tau+1}^n g(T_i | \theta_2) + (N - \tau) \log [1 - F(T_\tau | \theta_1)] \\ &+ (N - n) \log [1 - G(T_n | \theta_2)]. \end{aligned} \quad (1.61)$$

If the parameter N is known in which where hardware reliability is commonly considered, for example, then the likelihood function is given by

$$\begin{aligned} L(\tau, N, \theta_1, \theta_2 | T_1, T_2, \dots, T_n) &+ \sum_{i=1}^{\tau} f(T_i | \theta_1) + \sum_{i=\tau+1}^n g(T_i | \theta_2) \\ &+ (N - \tau) \log [1 - F(T_\tau | \theta_1)] + (N - n) \\ &\log [1 - G(T_n | \theta_2)]. \end{aligned}$$

The maximum likelihood estimator (MLE) of the change-point value $\hat{\tau}$ and $(\hat{N}, \hat{\theta}_1, \hat{\theta}_2)$ can be obtained by taking partial derivatives of the log likelihood function in (1.61) with respect to the unknown parameters that maximize the function. It should be noted that there is no closed form for $\hat{\tau}$, but it can be obtained by calculating the log likelihood for each possible value of τ , $1 \leq \tau \leq (n - 1)$, and selecting the value that maximizes the log likelihood function.

Application 2: A Software Model with a Change Point

In this application, we examine the case where the sample size N is unknown. Consider a software reliability model developed by *Jelinski* and *Moranda* in 1972, often called the Jelinski–Moranda model. The assumptions of the model are as follows:

1. There are N initial faults in the program.
2. A detected fault is removed instantaneously and no new fault is introduced.
3. Each failure caused by a fault occurs independently and randomly in time according to an exponential distribution.
4. The functions F and G are exponential distributions with failure rate parameters λ_1 and λ_2 , respectively.

Based on these assumptions, the inter-failure times X_1, X_2, \dots, X_n are independently exponentially distributed. Specifically, $X_i = T_i - T_{i-1}$, $i = 1, 2, \dots, \tau$, are exponentially distributed with parameter $\lambda_1(N - i + 1)$, where λ_1 is the initial fault detection rate of the first τ failures, and $X_j = T_j - T_{j-1}$, $j = \tau + 1, \tau + 2, \dots, n$ are exponentially distributed with parameter $\lambda_2(N - \tau - j + 1)$, where λ_2 is the fault detection rate of the first $n - \tau$ failures. If $\lambda_1 = \lambda_2$,

it means that each fault removal is the same and that the change-point model becomes the Jelinski–Moranda software reliability model [7].

The MLEs of the parameters $(\tau, N, \lambda_1, \lambda_2)$ can be obtained by solving the following equations simultaneously:

$$\hat{\lambda}_1 = \frac{\tau}{\sum_{i=1}^{\tau} (\hat{N} - i + 1) x_i}, \quad (1.62)$$

$$\hat{\lambda}_2 = \frac{(n - \tau)}{\sum_{i=\tau+1}^n (\hat{N} - i + 1) x_i}, \quad (1.63)$$

$$\sum_{i=1}^n \frac{1}{(\hat{N} - i + 1)} = \hat{\lambda}_1 \sum_{i=1}^{\tau} x_i + \hat{\lambda}_2 \sum_{i=\tau+1}^n x_i. \quad (1.64)$$

To illustrate the model, we use the dataset shown in Table 1.3 to obtain the unknown parameters $(\tau, N, \lambda_1, \lambda_2)$ using (1.62)–(1.64). The data in Table 1.3 [8] shows the successive inter-failure times for a real-time command and control system. The table reads from left to right in rows, and the recorded times are execution times, in seconds. There are 136 failures in total. Figure 1.8 plots the log-likelihood function versus the number of failures. The MLEs of the parameters $(\tau, N, \lambda_1, \lambda_2)$ with one change point are given by

$$\hat{\tau} = 16, \hat{N} = 145, \hat{\lambda}_1 = 1.1 \times 10^{-4}, \\ \hat{\lambda}_2 = 0.31 \times 10^{-4}.$$

If we do not consider a change point in the model, the MLEs of the parameters N and λ , can be given as

$$\hat{N} = 142, \hat{\lambda} = 0.35 \times 10^{-4}.$$

From Fig. 1.8, it is clear that it is worth considering change points in reliability functions.

Table 1.3 Successive inter-failure times (in s) for a real-time command system

3	30	113	81	115	9	2	91	112	15
138	50	77	24	108	88	670	120	26	114
325	55	242	68	422	180	10	1146	600	15
36	4	0	8	227	65	176	58	457	300
97	263	452	255	197	193	6	79	816	1351
148	21	233	134	357	193	236	31	369	748
0	232	330	365	1222	543	10	16	529	379
44	129	810	290	300	529	281	160	828	1011
445	296	1755	1064	1783	860	983	707	33	868
724	2323	2930	1461	843	12	261	1800	865	1435
30	143	108	0	3110	1247	943	700	875	245
729	1897	447	386	446	122	990	948	1082	22
75	482	5509	100	10	1071	371	790	6150	3321
1045		648	5485	1160	1864	4116			

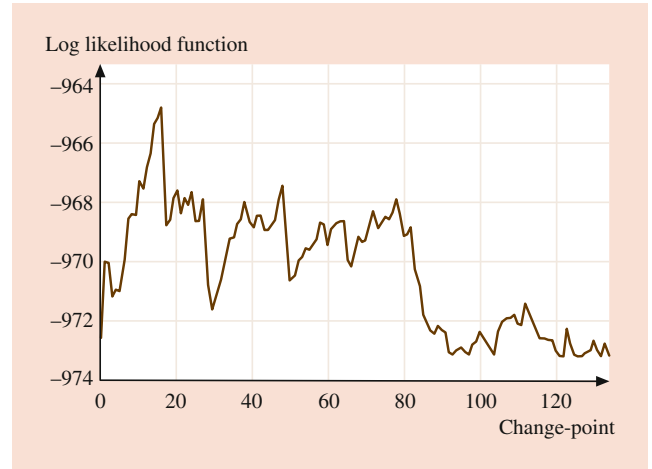


Fig. 1.8 The log likelihood function versus the number of failures

1.4.4 Goodness-of-Fit Techniques

The problem discussed here is one of comparing an observed sample distribution with a theoretical distribution. Practitioners often wonder how to test some hypothesis about the distribution of a population. If the test is concerned with the agreement between the distribution of a set of observed sample values and a theoretical distribution, we call it a test of “goodness of fit.”

The basic question in validating distribution models is whether the shape of the fitted model corresponds to that of the data. To do that, we may just simply make a direct comparison of the observed data with what we expect to see from the fitted distribution model. Two common techniques that will be discussed are the χ^2 goodness-of-fit test and the Kolmogorov–Smirnov “ d ” test.

Chi-Squared Test

The chi-square test often requires large samples and is applied by comparing the observed frequency distribution of the sample to the expected value under the assumption of the distribution. More specifically, consider a large sample of size N . Let $a_0 < a_1 < \dots < a_k$ be the upper points of k subintervals of the frequency distribution. The following statistic

$$\chi^2 = \sum_{i=1}^k \left(\frac{x_i - \mu_i}{\sigma_i} \right)^2 \quad (1.65)$$

has a chi-squared (χ^2) distribution with k degrees of freedom. The procedure used for the chi-squared test is as follows:

1. Divide the sample data into mutually exclusive cells (normally 8–12) such that the range of the random variable is covered.
2. Determine the frequency, f_i , of sample observations in each cell.

3. Determine the theoretical frequency, F_i , for each cell (the area under density function between cell boundaries X_n – total sample size). Note that the theoretical frequency for each cell should be greater than 1. This step normally requires estimates for the population parameters, which can be obtained from the sample data.
4. Form the statistic

$$A = \sum_{i=1}^k \frac{(f_i - F_i)^2}{F_i} \tag{1.66}$$

5. From the χ^2 tables, choose a value of χ^2 with the desired significance level and degrees of freedom ($= k - 1 - r$, where r is the number of population parameters estimated).
6. Reject the hypothesis that the sample distribution is the same as the theoretical distribution if

$$A > \chi^2_{1-\alpha, k-1-r}$$

where α is called the significance level.

Example 1.21 Given the data in Table 1.4, can the data be represented by the exponential distribution with a significance level of α ?

From the above calculation, $\hat{\lambda} = 0.00263$, $R_i = e^{-\lambda t_i}$ and $Q_i = 1 - R_i$. Given that the significance level α is 0.1, from (1.66) we obtain

$$A = \sum_{i=1}^{11} \frac{(f_i - F_i)^2}{F_i} = 6.165.$$

From Table 1.11 in Appendix A, the value of χ^2 with nine degrees of freedom and $\alpha = 0.1$ is 14.68; that is,

$$\chi^2_{9,0.1} = 14.68.$$

Since $S = 6.165 < 14.68$, we would not reject the hypothesis of an exponential with $\lambda = 0.00263$.

Table 1.4 Sample observations for each cell boundary

Cell boundaries	f_i	$Q_i = (1 - R_i) 60$	$F_i = Q_i - Q_{i-1}$
0–100	10	13.86	13.86
100–200	9	24.52	10.66
200–300	8	32.71	8.19
300–400	8	39.01	6.30
400–500	7	43.86	4.85
500–600	6	47.59	3.73
600–700	4	50.45	2.86
700–800	4	52.66	2.21
800–900	2	54.35	1.69
900–1000	1	55.66	1.31
>1000	1	58.83	2.17

If in the statistic

$$A = \sum_{i=1}^k \left(\frac{f_i - F_i}{\sqrt{F_i}} \right)^2, \quad \left(\frac{f_i - F_i}{\sqrt{F_i}} \right)$$

is approximately normal for large samples, then A also has a χ^2 distribution. This is the basis for the goodness-of-fit test.

Kolmogorov–Smirnov (KS) Test

Both the χ^2 and “KS” tests are nonparametric tests. However, the χ^2 test largely assumes sample normality of the observed frequency about its mean, while “KS” assumes only a continuous distribution. Let $X_1 \leq X_2 \leq X_3 \leq \dots \leq X_n$ denote the ordered sample values. Define the observed distribution function, $F_n(x)$, as:

$$F_n(X) = \begin{cases} 0 & \text{for } x \leq x_1 \\ \frac{i}{n} & \text{for } x_i < x \leq x_{i+1} \\ 1 & \text{for } x > x_n \end{cases}$$

Assume the testing hypothesis

$$H_0 : F(x) = F_0(x),$$

where $F_0(x)$ is a given continuous distribution and $F(x)$ is an unknown distribution. Let

$$d_n = \sup_{-\infty < x < \infty} |F_n(x) - F_0(x)| \tag{1.67}$$

Since $F_0(x)$ is a continuous increasing function, we can evaluate $|F_n(x) - F_0(x)|$ for each n . If $d_n \leq d_{n,\alpha}$, then we will not reject the hypothesis H_0 ; otherwise, we will reject it when $d_n > d_{n,\alpha}$. The value of $d_{n,\alpha}$ can be found in Table 1.7 in Appendix A, where n is the sample size and α is the level of significance.

Example 1.22 Determine whether the following failure data (in days) of a system:

10.50, 1.75, 6.10, 1.30, 15.00, 8.20, 0.50, 20.50, 11.05, 4.60

be represented as a sample from an exponential population distribution with constant rate $\lambda = 0.20$ failures per day at the ($\alpha =$) 5% level of the significance using KS test.

Under the hypothesis that failure times are exponential distribution, so the theoretical pdf and cdf are given by:

$$f(x) = 0.2e^{-0.2x} \quad \text{for } x > 0$$

and

$$F(x) = 1 - e^{-0.2x}$$

Table 1.5 The observed and theoretical cdf values

Failure time x	$F_n(x)$	$F_0(x)$	$ F_n(x) - F_0(x) $
0.50	0.10	0.0952	0.0048
1.30	0.20	0.2289	0.0289
1.75	0.30	0.2953	0.0047
4.60	0.40	0.6015	0.2015
6.10	0.50	0.7048	0.2048
8.20	0.60	0.8061	0.2061
10.50	0.70	0.8776	0.1776
11.05	0.80	0.8903	0.0903
15.00	0.90	0.9503	0.0503
20.50	1.00	0.9835	0.0165

respectively. From Table 1.5, the maximum difference D_n is 0.2061.

From Table 1.7 in Appendix A, here $n = 10$ and $\alpha = 5\%$, then we obtain the critical value $d_{n,\alpha} = 0.409$. Since $D_n \leq d_{n,\alpha}$, therefore the null hypothesis, that failure times are exponentially distributed with constant rate $\lambda = 0.20$, cannot be rejected at the 5% level of significance.

1.4.5 Least Squares Estimation

One common approach to curve fitting, which is unrelated to normal regression theory and MLE estimates of coefficients but uses identical formulae, is called the method of least squares. This method is based on minimizing the sum of the squared distances from the best fit line to the actual data points. It just so happens that finding the MLEs for the coefficients of the regression line also involves this sum of squared distances.

Normal Linear Regression

Regression considers the distribution of one variable as a function of another when the other variable is fixed at each of several levels. In the normal bivariate case, consider the distribution of X as a function of given values of Z where $X = \alpha + \beta Z$. Consider a sample of n observations (x_i, z_i) . We can obtain the likelihood and its natural log for the normal distribution as follows:

$$\begin{aligned}
 f(x_1, x_2, \dots, x_n) &= \frac{1}{2\pi^{n/2}} \left(\frac{1}{\sigma^2} \right)^{n/2} e^{-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \alpha - \beta z_i)^2} \\
 \ln L &= -\frac{n}{2} \log 2\pi - \frac{n}{2} \log \sigma^2 \\
 &\quad - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \alpha - \beta z_i)^2.
 \end{aligned}$$

Taking the partial derivatives of $\ln L$ with respect to α and β , we have

$$\begin{aligned}
 \frac{\partial \ln L}{\partial \alpha} &= \sum_{i=1}^n (x_i - \alpha - \beta z_i) = 0, \\
 \frac{\partial \ln L}{\partial \beta} &= \sum_{i=1}^n z_i (x_i - \alpha - \beta z_i) = 0.
 \end{aligned}$$

The solutions to the simultaneous equations are

$$\begin{aligned}
 \hat{\alpha} &= \bar{X} - \beta \bar{Z}, \\
 \hat{\beta} &= \frac{\sum_{i=1}^n (X_i - \bar{X})(Z_i - \bar{Z})}{\sum_{i=1}^n (Z_i - \bar{Z})^2}. \tag{1.68}
 \end{aligned}$$

Least Squared Straight Line Fit

Assume that there is a linear relationship between X and $E(Y | x)$; that is, that $E(Y | x) = a + bx$. Given a set of data, we want to estimate the coefficients a and b that minimize the sum of the squares. Suppose that the desired polynomial, $p(x)$, is written as

$$\sum_{i=0}^m a_i x^i,$$

where a_0, a_1, \dots, a_m are to be determined. The method of least squares chooses as ‘‘solutions’’ those coefficients that minimize the sum of the squares of the vertical (y) distances from the data points to the presumed polynomial. This means that the ‘‘best’’ polynomial is the one whose coefficients minimize the function L , where

$$L = \sum_{i=1}^n [y_i - p(x_i)]^2.$$

Here, we will only treat the linear case, where $X = \alpha + \beta Z$. The procedure for higher-order polynomials is identical, although the computations become much more tedious. Assume a straight line of the form $X = \alpha + \beta Z$. For each observation (x_i, z_i) : $X_i = \alpha + \beta Z_i$, let

$$Q = \sum_{i=1}^n (x_i - \alpha - \beta z_i)^2.$$

We wish to find estimates for α and β that minimize Q . Taking partial differentials, we obtain

$$\begin{aligned}
 \frac{\partial Q}{\partial \alpha} &= -2 \sum_{i=1}^n (x_i - \alpha - \beta z_i) = 0, \\
 \frac{\partial Q}{\partial \beta} &= -2 \sum_{i=1}^n z_i (x_i - \alpha - \beta z_i) = 0.
 \end{aligned}$$

Note that the above are the same as the MLE equations for normal linear regression. Therefore, we obtain the following results:

$$\hat{\alpha} = \bar{x} - \beta \bar{z},$$

$$\hat{\beta} = \frac{\sum_{i=1}^n (x_i - \bar{x})(z_i - \bar{z})}{\sum_{i=1}^n (z_i - \bar{z})^2}. \quad (1.69)$$

The above gives an example of least squares applied to a linear case. The same pattern applies for higher-order curves with 3, 4, and so on solutions.

1.4.6 Interval Estimation

A point estimate is sometimes inadequate at providing an estimate for an unknown parameter, since it rarely coincides with the true value of the parameter. An alternative way is to obtain a confidence interval estimation of the form $[\theta_L, \theta_U]$ where θ_L is the lower bound and θ_U is the upper bound.

Point estimates can become more useful if some measure of their error is given; in other words, if some kind of tolerance for their high and low values is given. Thus, if an interval estimator is $[\theta_L, \theta_U]$ with a given probability $(1 - \alpha)$, then θ_L and θ_U are called the 100 $(1 - \alpha)\%$ confidence limits for the given parameter θ , and the interval between them is a 100 $(1 - \alpha)\%$ confidence interval, while $(1 - \alpha)$ is called the confidence coefficient.

Confidence Intervals for Normal Parameters

The one-dimensional normal distribution has two parameters: mean μ and variance σ^2 . The simultaneous employment of both parameters in a confidence statement concerning percentages of the population will be discussed in the next section on tolerance limits. Hence, individual confidence statements about μ and σ^2 will be discussed here.

Confidence Limits for a Mean μ with a Known σ^2

It is easy to show that the statistic

$$Z = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}}$$

is a standard normal distribution, where

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i.$$

Hence, a 100 $(1 - \alpha)\%$ confidence interval for the mean μ is given by

$$P \left[\bar{X} - Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} < \mu < \bar{X} + Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \right] = 1 - \alpha. \quad (1.70)$$

In other words,

$$\mu_L = \bar{X} - Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \quad \text{and} \quad \mu_U = \bar{X} + Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}.$$

Example 1.23 Draw a sample of size 4 from a normal distribution with a known variance = 9, say $x_1 = 2$, $x_2 = 3$, $x_3 = 5$, and $x_4 = 2$. Determine the location of the true mean (μ). The sample mean can be calculated as

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} = \frac{2 + 3 + 5 + 2}{4} = 3.$$

Assuming that $\alpha = 0.05$ and, from the standard normal distribution (Table 1.8 in Appendix A), $Z_{0.025} = 1.96$, then we obtain

$$P \left[3 - 1.96 \frac{3}{\sqrt{4}} < \mu < 3 + 1.96 \frac{3}{\sqrt{4}} \right] = 0.95,$$

$$P [0.06 < \mu < 5.94] = 0.95.$$

This example shows that there is a 95% probability that the true mean is somewhere between 0.06 and 5.94. Now, μ is a fixed parameter and does not vary, so how do we interpret the probability? If samples of size 4 were to be repeatedly drawn, a different set of limits would be constructed each time. If this is the case, the interval becomes the random variable and the interpretation is that, for 95% of the time, the interval constructed in this way will contain the true (fixed) parameter.

Confidence Limits for a Mean μ with an Unknown σ^2

Let

$$S = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2}. \quad (1.71)$$

It can be shown that the statistic

$$T = \frac{\bar{X} - \mu}{\frac{S}{\sqrt{n}}}$$

has a t distribution with $(n - 1)$ degrees of freedom (see Table 1.9 in Appendix A). Thus, for a given sample mean and sample standard deviation, we obtain

$$P [|T| < t_{\frac{\alpha}{2}, n-1}] = 1 - \alpha.$$

Hence, a 100 $(1 - \alpha)\%$ confidence interval for the mean μ is given by

$$P \left[\bar{X} - t_{\frac{\alpha}{2}, n-1} \frac{S}{\sqrt{n}} < \mu < \bar{X} + t_{\frac{\alpha}{2}, n-1} \frac{S}{\sqrt{n}} \right] = 1 - \alpha. \quad (1.72)$$

Example 1.24 The variability of a new product was investigated. An experiment was run using a sample of size $n = 25$; the sample mean was found to be $\bar{X} = 50$ and the sample variance $S^2 = 16$. From Table 1.9 in Appendix A, $t_{\frac{\alpha}{2}, n-1} = t_{0.025, 24} = 2.064$. The 95% confidence limit for μ is given by

$$P \left[50 - 2.064 \sqrt{\frac{16}{25}} < \mu < 50 + 2.064 \sqrt{\frac{16}{25}} \right] = 0.95, \\ P [48.349 < \mu < 51.651] = 0.95.$$

Note that, for one-sided limits, one should choose t_α , or $t_{1-\alpha}$.

Confidence Limits on σ^2

Note that $n\hat{\sigma}^2/\sigma^2$ has a χ^2 distribution with $(n-1)$ degrees of freedom. Correcting for the bias in $\hat{\sigma}^2$, it is clear that $(n-1)\hat{\sigma}^2/\sigma^2$ has this same distribution. Hence,

$$P \left[\chi_{\frac{\alpha}{2}, n-1}^2 < \frac{(n-1)S^2}{\sigma^2} < \chi_{1-\frac{\alpha}{2}, n-1}^2 \right] = 1 - \alpha$$

or

$$P \left[\frac{\sum (x_i - \bar{x})^2}{\chi_{1-\frac{\alpha}{2}, n-1}^2} < \sigma^2 < \frac{\sum (x_i - \bar{x})^2}{\chi_{\frac{\alpha}{2}, n-1}^2} \right] = 1 - \alpha. \quad (1.73)$$

Again, for one-sided limits, one should choose $\chi^2(\alpha)$ or $\chi^2(1-\alpha)$.

Confidence Intervals for Exponential Parameters

The pdf and cdf for the exponential distribution are

$$f(x) = \lambda e^{-\lambda x}, \quad x > 0, \lambda > 0$$

and

$$F(x) = 1 - e^{-\lambda x},$$

respectively. It was shown that the distribution of a function of the estimate

$$\hat{\lambda} = \frac{r}{\sum_{i=1}^n x_i + (n-r)x_r} \quad (1.74)$$

derived from a test of n identical components with common exponential failure density (failure rate λ), whose testing was stopped after the r th failure, was chi-squared (χ^2), i.e.,

$$2r \frac{\lambda}{\hat{\lambda}} = 2\lambda T \quad (\chi^2 \text{ distribution with } 2r \text{ degrees of freedom}),$$

where T is the total time accrued by all units. Knowing the distribution of $2\lambda T$ allows us to obtain the confidence limits on the parameter as:

$$P \left[\chi_{1-\frac{\alpha}{2}, 2r}^2 < 2\lambda T < \chi_{\frac{\alpha}{2}, 2r}^2 \right] = 1 - \alpha$$

or, equivalently, that

$$P \left[\frac{\chi_{1-\frac{\alpha}{2}, 2r}^2}{2T} < \lambda < \frac{\chi_{\frac{\alpha}{2}, 2r}^2}{2T} \right] = 1 - \alpha.$$

This means that in $(1-\alpha)\%$ of the samples of a given size n , the random interval

$$\left(\frac{\chi_{1-\frac{\alpha}{2}, 2r}^2}{2T}, \frac{\chi_{\frac{\alpha}{2}, 2r}^2}{2T} \right)$$

will contain a population of constant failure rate. For $\theta = 1/\lambda$ or the MTBF, the above confidence limits change to

$$P \left[\frac{2T}{\chi_{\frac{\alpha}{2}, 2r}^2} < \theta < \frac{2T}{\chi_{1-\frac{\alpha}{2}, 2r}^2} \right] = 1 - \alpha.$$

If testing is stopped at a fixed time rather than a fixed number of failures, the number of degrees of freedom in the lower limit increases by two. Table 1.6 shows the confidence limits for θ , the mean of the exponential density.

Confidence Intervals for Binomial Parameters

Consider a sequence of n Bernoulli trials with k successes and $(n-k)$ failures. We now determine one-sided upper and lower and two-sided limits on the parameter p , the probability of success. For the lower limit, the binomial sum is set up such that the probability of k or more successes with a true p as low as p_L is only $\alpha/2$. This means that the probability of k or more successes with a true p higher than p_L is $(1 - \frac{\alpha}{2})$, such that

Table 1.6 Confidence limits for θ

Confidence limits	Fixed number of failures	Fixed time
One-sided lower limit	$\frac{2T}{\chi_{\alpha, 2r}^2}$	$\frac{2T}{\chi_{\alpha, 2r+2}^2}$
One-sided upper limit	$\frac{2T}{\chi_{1-\alpha, 2r}^2}$	$\frac{2T}{\chi_{1-\alpha, 2r}^2}$
Two-sided limits	$\frac{2T}{\chi_{\alpha/2, 2r}^2}, \frac{2T}{\chi_{1-\alpha/2, 2r}^2}$	$\frac{2T}{\chi_{\alpha/2, 2r+2}^2}, \frac{2T}{\chi_{1-\alpha/2, 2r}^2}$

$$\sum_{i=k}^n \binom{n}{i} p_L^i (1-p_L)^{n-i} = \frac{\alpha}{2}.$$

Similarly, the binomial sum for the upper limit is

$$\sum_{i=k}^n \binom{n}{i} p_U^i (1-p_U)^{n-i} = \frac{\alpha}{2}$$

or, equivalently,

$$\sum_{i=0}^{k-1} \binom{n}{i} p_U^i (1-p_U)^{n-i} = \frac{\alpha}{2}.$$

Solving for p_L and p_U in the above equations,

$$P[p_L < p < p_U] = 1 - \alpha.$$

For one-sided limits, merely change $\alpha/2$ to α .

Example 1.25 Given $n = 100$ with 25 successes, and 75 failures, an 80% two-sided confidence limit on p can be obtained as follows:

$$\begin{aligned} \sum_{i=25}^{100} \binom{100}{i} p_L^i (1-p_L)^{100-i} &= 0.10, \\ \sum_{i=0}^{24} \binom{100}{i} p_U^i (1-p_U)^{100-i} &= 0.10. \end{aligned}$$

Solving the above two equations simultaneously, we obtain

$$\begin{aligned} p_L &\approx 0.194 \text{ and } p_U \approx 0.313, \\ P[0.194 < p < 0.313] &= 0.80. \end{aligned}$$

Continuing with Example 1.25 above, we now find an 80% one-sided confidence limit on p .

We start by setting the top equation to 0.20 and solving for p_L . It is then easy to obtain $p_L = 0.211$ and $P[p > 0.211] = 0.80$.

Let us now define $\bar{p} = k/n$, the number of successes divided by the number of trials. For large values of n and if $np > 5$ and $n(1-p) > 5$, and from the central limit theorem [9], the statistic

$$Z = \frac{(\bar{p} - p)}{\sqrt{\frac{\bar{p}(1-\bar{p})}{n}}}$$

approximates to the standard normal distribution. Hence

$$P[-z_{\frac{\alpha}{2}} < Z < z_{\frac{\alpha}{2}}] = 1 - \alpha.$$

Then

$$\begin{aligned} P\left[\bar{p} - z_{\frac{\alpha}{2}} \sqrt{\frac{\bar{p}(1-\bar{p})}{n}} < p \right. \\ \left. < \bar{p} + z_{\frac{\alpha}{2}} \sqrt{\frac{\bar{p}(1-\bar{p})}{n}}\right] = 1 - \alpha. \end{aligned}$$

Example 1.26 Find the two-sided confidence limit for $n = 900$, $k = 180$, and $\alpha = 0.05$. Then we obtain $p = 180/900 = 0.2$ and

$$\begin{aligned} P\left[0.2 - 1.96 \sqrt{\frac{0.2(0.8)}{900}} < p \right. \\ \left. < 0.2 + 1.96 \sqrt{\frac{0.2(0.8)}{900}}\right] = 0.95, \\ P[0.174 < p < 0.226] &= 0.95. \end{aligned}$$

Confidence Intervals for Poisson Parameters

Limits for the Poisson parameters are completely analogous to those for the binomial distribution except that the sample space is denumerable instead of finite. The lower and upper limits can be solved simultaneously in the following equations:

$$\begin{aligned} \sum_{i=k}^{\infty} \frac{\lambda_L^i e^{-\lambda_L}}{i!} &= \frac{\alpha}{2}, \\ \sum_{i=k}^{\infty} \frac{\lambda_U^i e^{-\lambda_U}}{i!} &= 1 - \frac{\alpha}{2}, \end{aligned}$$

or, equivalently,

$$\begin{aligned} \sum_{i=k}^{\infty} \frac{\lambda_L^i e^{-\lambda_L}}{i!} &= \frac{\alpha}{2}, \\ \sum_{i=0}^{k-1} \frac{\lambda_U^i e^{-\lambda_U}}{i!} &= \frac{\alpha}{2}. \end{aligned}$$

The one-sided limits are constructed in the same way as for binomial limits.

1.4.7 Nonparametric Tolerance Limits

Nonparametric tolerance limits are based on the smallest and largest observation in the sample, designated X_S and X_L , respectively. Due to their nonparametric nature, these limits are quite insensitive, and obtaining precisions similar to the parametric methods necessitates much larger samples. An interesting question here is to determine the sample size required to include at least 100 $(l - \alpha)\%$ of the population between X_S and X_L with a given probability y .

For two-sided tolerance limits, if $(1 - \alpha)$ is the minimum proportion of the population contained between the largest

observation X_L and the smallest observation X_S with a confidence $(1 - \gamma)$, then it can be shown that

$$n(1 - \alpha)^{n-1} - (n - 1)(1 - \alpha)^n = \gamma.$$

Therefore, the number of observations required is given by

$$n = \left(\frac{(2 - \alpha)}{4\alpha} \chi_{1-\gamma,4}^2 + \frac{1}{2} \right) + 1, \quad (1.75)$$

where a value of $\chi_{1-\gamma,4}^2$ is given in Table 1.11 in Appendix A.

Example 1.27 Determine the tolerance limits that include at least 90% of the population with probability 0.95. Here,

$$\alpha = 0.1, \quad \gamma = 0.95 \quad \text{and} \quad \chi_{0.05,4}^2 = 9.488.$$

Therefore, a sample of size

$$n = \left[\frac{(2 - 0.1)}{4(0.1)} (9.488) + \frac{1}{2} \right] + 1 = 46$$

is required. For a one-sided tolerance limit, the number of observations required is given by

$$n = \left(\frac{\log(1 - \gamma)}{\log(1 - \alpha)} \right) + 1.$$

Example 1.28 As in Example 1.27, we wish to find a lower tolerance limit; that is, the number of observations required such that the probability is 0.95 that at least 90% of the population will exceed X_S . This is given by

$$n = \left(\frac{\log(1 - 0.95)}{\log(1 - 0.1)} \right) + 1 = 30.$$

One can easily generate a table containing the sample size required to include a given percentage of the population between X_S and X_L with a given confidence, or the sample size required to include a given percentage of the population above or below X_S or X_L , respectively.

1.4.8 Sequential Sampling

A sequential sampling scheme is one in which items are drawn one at a time and the results at any stage determine whether sampling or testing should stop. Thus, any sampling procedure for which the number of observations is a random variable can be regarded as sequential sampling. Sequential tests derive their name from the fact that the sample size is not determined in advance, but allowed to “float” with a decision (accept, reject, or continue test) after each trial or data point.

In general, let us consider the hypothesis

$$H_0 : f(x) = f_0(x) \quad \text{versus} \quad H_1 : f(x) = f_1(x).$$

For an observational test, say X_1 , if $X_1 \leq A$, then we will accept the testing hypothesis [$H_0: f(x) = f_0(x)$]; if $X_1 \geq A$ we will reject H_0 and accept $H_1: f(x) = f_1(x)$. Otherwise, we will continue to perform at least one more test. The interval $X_1 \leq A$ is called the acceptance region. The interval $X_1 \geq A$ is called the rejection or critical region.

A “good” test is one that makes the α and β errors as small as possible where

$$P \{\text{Type I error}\} = P \{\text{Reject } H_0 | H_0 \text{ is True}\} = \alpha$$

$$P \{\text{Type II error}\} = P \{\text{Accept } H_0 | H_0 \text{ is False}\} = \beta$$

However, there is not much freedom to do this without increasing the sample size. A common procedure is to fix the β error and then choose a critical region to minimize the error or to maximize the “power” (power = $1 - \beta$) of the test, or to choose the critical region so as to equalize the α and β errors to reasonable levels.

One criterion (similar to the MLE) used to construct tests is called the “probability ratio,” which is the ratio of the sample densities under H_1/H_0 . Consider the ratio of probabilities

$$\lambda_m = \frac{\prod_{i=1}^n f_1(x_i)}{\prod_{i=1}^n f_0(x_i)} > k. \quad (1.76)$$

Here, x_1, x_2, \dots, x_n are n independent random observations and k is chosen to give the desired a error.

Recall from the MLE discussion in Sect. 1.4.1 that $f_1(x_1), f_1(x_2), \dots, f_1(x_n)$ are maximized under H_1 when the parameter(s), e.g., $\theta = \theta_1$ and similarly $f_0(x_1), f_0(x_2), \dots, f_0(x_n)$ are maximized when $\theta = \theta_0$. Thus, the ratio will become large if the sample favors H_1 and will become small if the sample favors H_0 . Therefore, the test will be called a sequential probability ratio test if we

1. Stop sampling and reject H_0 as soon as $\lambda_m \geq A$
2. Stop sampling and accept H_0 as soon as $\lambda_m \leq B$
3. Continue sampling as long as $B < \lambda_m < A$, where $A > B$

The selection of A and B using the above test, as suggested by Wald (see [9]), can be determined as follows:

$$B = \frac{\beta}{1 - \alpha} \quad \text{and} \quad A = \frac{1 - \beta}{\alpha}$$

The bases for α and β are therefore:

$$P[\lambda_m > A | H_0] = \alpha$$

$$P[\lambda_m < A | H_1] = \beta$$

1.4.9 Bayesian Methods

The Bayesian approach to statistical inference is based on a theorem first presented by Thomas Bayes. To demonstrate the approach, let X have a pdf $f(x)$, which is dependent on θ . In the traditional statistical inference approach, θ is an unknown parameter, and hence is a constant. We now describe our prior supposition for the value of θ by a pdf of $h(\theta)$. This amounts to quantitatively assessing subjective judgment and should not be confused with the so-called objective probability assessment derived from the long-term frequency approach. Thus, θ will now essentially be treated as a random variable θ with a pdf of $h(\theta)$.

Consider a random sample X_1, X_2, \dots, X_n from $f(x)$ and define a statistic Y as a function of this random sample. Then there exists a conditional pdf $g(y | \theta)$ of Y for a given θ . The joint pdf for y and θ is

$$f(\theta, y) = h(\theta)g(y|\theta).$$

If θ is continuous, then

$$f_1(y) = \int_{\theta} h(\theta)g(y|\theta)d\theta$$

is the marginal pdf for the statistic y . Given the information y , the conditional pdf for θ is

$$k(\theta|y) = \frac{h(\theta)g(y|\theta)}{f_1(y)} \quad \text{for } f_1(y) > 0$$

$$= \frac{h(\theta)g(y|\theta)}{\int_{\theta} h(\theta)g(y|\theta)d\theta}$$

If θ is discrete, then

$$f_1(y) = \sum_k P(\theta_k)P(y|\theta_k)$$

and

$$P(\theta_i|y_i) = \frac{P(\theta_k)P(y_i|\theta_i)}{\sum_k P(\theta_k)P(y_j|\theta_k)}$$

where $P(\theta_j)$ is the prior probability of event θ_i and $P(\theta_j | y_j)$ is the posterior probability of event y_j given θ_i . This is simply a form of Bayes' theorem. Here, $h(\theta)$ is the prior pdf that expresses our belief about the value of θ before the data ($Y = y$) became available. Then $k(\theta | y)$ is the posterior pdf, given the data ($Y = y$).

Note that the difference in the shape of the prior pdf $h(\theta)$ compared to the posterior pdf $k(\theta | y)$ due to the information

is a result of the product of $g(y | \theta)$ and $h(\theta)$, because $f_1(y)$ is simply a normalization constant for a fixed y . The idea of reliability is to take "prior" data and combine it with current data to gain a better estimate or confidence interval or test than would be possible with either on their own. As more current data is acquired, the prior data is "washed out" [1].

1.5 Further Reading

The reader interested in a deeper understanding of advanced probability theory and stochastic processes should note the following citations, which refer to highly recommended books: [9–13].

Appendix A: Distribution Tables (Tables 1.7, 1.8, 1.9, 1.10, and 1.11)

Table 1.7 Critical values $d_{n,\alpha}$ for the Kolmogorov–Smirnov test

n/α	0.2	0.1	0.05	0.02	0.01
1	0.900	0.950	0.975	0.990	0.995
2	0.684	0.776	0.842	0.900	0.929
3	0.565	0.636	0.708	0.785	0.829
4	0.493	0.565	0.624	0.689	0.734
5	0.447	0.509	0.563	0.627	0.669
6	0.410	0.468	0.519	0.577	0.617
7	0.381	0.436	0.483	0.538	0.576
8	0.358	0.410	0.454	0.507	0.542
9	0.339	0.387	0.430	0.480	0.513
10	0.323	0.369	0.409	0.457	0.489
11	0.308	0.352	0.391	0.437	0.468
12	0.296	0.338	0.375	0.419	0.449
13	0.285	0.325	0.361	0.404	0.432
14	0.275	0.314	0.349	0.390	0.418
15	0.266	0.304	0.338	0.377	0.404
16	0.258	0.295	0.327	0.366	0.392
17	0.250	0.286	0.318	0.355	0.381
18	0.244	0.279	0.309	0.346	0.371
19	0.237	0.271	0.301	0.337	0.361
20	0.232	0.265	0.294	0.329	0.352
21	0.226	0.259	0.287	0.321	0.344
22	0.221	0.253	0.281	0.314	0.337
23	0.216	0.247	0.275	0.307	0.330
24	0.212	0.242	0.264	0.301	0.323
25	0.208	0.238	0.264	0.295	0.317
26	0.204	0.233	0.259	0.290	0.311
27	0.200	0.229	0.254	0.284	0.305
28	0.197	0.225	0.250	0.279	0.300
29	0.193	0.221	0.246	0.275	0.295
30	0.190	0.218	0.242	0.270	0.281

Table 1.8 Cumulative areas under the standard normal distribution

Z	0	1	2	3	4	5	6	7	8	9
-3.0	0.0013	0.0010	0.0007	0.0005	0.0003	0.0002	0.0002	0.0001	0.0001	0.0000
-2.9	0.0019	0.0018	0.0017	0.0017	0.0016	0.0016	0.0015	0.0015	0.0014	0.0014
-2.8	0.0026	0.0025	0.0024	0.0023	0.0023	0.0022	0.0021	0.0021	0.0020	0.0019
-2.7	0.0035	0.0034	0.0033	0.0032	0.0031	0.0030	0.0029	0.0028	0.0027	0.0026
-2.6	0.0047	0.0045	0.0044	0.0043	0.0041	0.0040	0.0039	0.0038	0.0037	0.0036
-2.5	0.0062	0.0060	0.0059	0.0057	0.0055	0.0054	0.0052	0.0051	0.0049	0.0048
-2.4	0.0082	0.0080	0.0078	0.0075	0.0073	0.0071	0.0069	0.0068	0.0066	0.0064
-2.3	0.0107	0.0104	0.0102	0.0099	0.0096	0.0094	0.0091	0.0089	0.0087	0.0084
-2.2	0.0139	0.0136	0.0132	0.0129	0.0126	0.0122	0.0119	0.0116	0.0113	0.0110
-2.1	0.0179	0.0174	0.0170	0.0166	0.0162	0.0158	0.0154	0.0150	0.0146	0.0143
-2.0	0.0228	0.0222	0.0217	0.0212	0.0207	0.0202	0.0197	0.0192	0.0188	0.0183
-1.9	0.0287	0.0281	0.0274	0.0268	0.0262	0.0256	0.0250	0.0244	0.0238	0.0233
-1.8	0.0359	0.0352	0.0344	0.0336	0.0329	0.0322	0.0314	0.0307	0.0300	0.0294
-1.7	0.0446	0.0436	0.0427	0.0418	0.0409	0.0401	0.0392	0.0384	0.0375	0.0367
-1.6	0.0548	0.0537	0.0526	0.0516	0.0505	0.0495	0.0485	0.0475	0.0465	0.0455
-1.5	0.0668	0.0655	0.0643	0.0630	0.0618	0.0606	0.0594	0.0582	0.0570	0.0559
-1.4	0.0808	0.0793	0.0778	0.0764	0.0749	0.0735	0.0722	0.0708	0.0694	0.0681
-1.3	0.0968	0.0951	0.0934	0.0918	0.0901	0.0885	0.0869	0.0853	0.0838	0.0823
-1.2	0.1151	0.1131	0.1112	0.1093	0.1075	0.1056	0.1038	0.1020	0.1003	0.0985
-1.1	0.1357	0.1335	0.1314	0.1292	0.1271	0.1251	0.1230	0.1210	0.1190	0.1170
-1.0	0.1587	0.1562	0.1539	0.1515	0.1492	0.1469	0.1446	0.1423	0.1401	0.1379
-0.9	0.1841	0.1814	0.1788	0.1762	0.1736	0.1711	0.1685	0.1660	0.1635	0.1611
-0.8	0.2119	0.2090	0.2061	0.2033	0.2005	0.1977	0.1949	0.1922	0.1894	0.1867
-0.7	0.2420	0.2389	0.2358	0.2327	0.2297	0.2266	0.2236	0.2206	0.2177	0.2148
-0.6	0.2743	0.2709	0.2676	0.2643	0.2611	0.2578	0.2546	0.2514	0.2483	0.2451
-0.5	0.3085	0.3050	0.3015	0.2981	0.2946	0.2912	0.2877	0.2843	0.2810	0.2776
-0.4	0.3446	0.3409	0.3372	0.3336	0.3300	0.3264	0.3228	0.3192	0.3156	0.3121
-0.3	0.3821	0.3783	0.3745	0.3707	0.3669	0.3632	0.3594	0.3557	0.3520	0.3483
-0.2	0.4207	0.4168	0.4129	0.4090	0.4052	0.4013	0.3974	0.3936	0.3897	0.3859
-0.1	0.4602	0.4562	0.4522	0.4483	0.4443	0.4404	0.4364	0.4325	0.4286	0.4247
-0.0	0.5000	0.4960	0.4920	0.4880	0.4840	0.4801	0.4761	0.4721	0.4681	0.4641
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7703	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9278	0.9292	0.9306	0.9319
1.5	0.9332	0.9345	0.9357	0.9370	0.9382	0.9394	0.9406	0.9418	0.9430	0.9441
1.6	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545
1.7	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633
1.8	0.9641	0.9648	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9700	0.9706

(continued)

Table 1.8 (continued)

Z	0	1	2	3	4	5	6	7	8	9
1.9	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.9750	0.9756	0.9762	0.9767
2.0	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817
2.1	0.9821	0.9826	0.9830	0.9834	0.9838	0.9842	0.9846	0.9850	0.9854	0.9857
2.2	0.9861	0.9864	0.9868	0.9871	0.9874	0.9878	0.9881	0.9884	0.9887	0.9890
2.3	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916
2.4	0.9918	0.9920	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936
2.5	0.9938	0.9940	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952
2.6	0.9953	0.9955	0.9956	0.9957	0.9959	0.9960	0.9961	0.9962	0.9963	0.9964
2.7	0.9965	0.9966	0.9967	0.9968	0.9969	0.9970	0.9971	0.9972	0.9973	0.9974
2.8	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.9980	0.9981
2.9	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986
3.0	0.9987	0.9990	0.9993	0.9995	0.9997	0.9998	0.9998	0.9999	0.9999	1.000

Table 1.9 Percentage points for the t -distribution ($t_{\alpha,r}$)

r/α	0.100	0.050	0.025	0.01	0.005	0.0025	0.001
1	3.078	6.314	12.706	31.821	63.657	127.32	318.310
2	1.886	2.920	4.303	6.965	9.925	14.089	23.326
3	1.638	2.353	3.182	4.541	5.841	7.453	10.213
4	1.533	2.132	2.776	3.747	4.604	5.598	7.173
5	1.476	2.015	2.571	3.365	4.032	4.773	5.893
6	1.440	1.943	2.447	3.143	3.707	4.317	5.208
7	1.415	1.895	2.365	2.998	3.499	4.029	4.785
8	1.397	1.860	2.306	2.896	3.355	3.833	4.501
9	1.383	1.833	2.262	2.821	3.250	3.690	4.297
10	1.372	1.812	2.228	2.764	3.169	3.581	4.144
11	1.363	1.796	2.201	2.718	3.106	3.497	4.025
12	1.356	1.782	2.179	2.681	3.055	3.428	3.930
13	1.350	1.771	2.160	2.650	3.012	3.372	3.852
14	1.345	1.761	2.145	2.624	2.977	3.326	3.787
15	1.341	1.753	2.131	2.602	2.947	3.286	3.733
16	1.337	1.746	2.120	2.583	2.921	3.252	3.686
17	1.333	1.740	2.110	2.567	2.898	3.222	3.646
18	1.330	1.734	2.101	2.552	2.878	3.197	3.610
19	1.328	1.729	2.093	2.539	2.861	3.174	3.579
20	1.325	1.725	2.086	2.528	2.845	3.153	3.552
21	1.323	1.721	2.080	2.518	2.831	3.135	3.527
22	1.321	1.717	2.074	2.508	2.819	3.119	3.505
23	1.319	1.714	2.069	2.500	2.807	3.104	3.485
24	1.318	1.711	2.064	2.492	2.797	3.091	3.467
25	1.316	1.708	2.060	2.485	2.787	3.078	3.450
26	1.315	1.706	2.056	2.479	2.779	3.067	3.435
27	1.314	1.703	2.052	2.473	2.771	3.057	3.421
28	1.313	1.701	2.048	2.467	2.763	3.047	3.408
29	1.311	1.699	2.045	2.462	2.756	3.038	3.396
30	1.310	1.697	2.042	2.457	2.750	3.030	3.385
40	1.303	1.684	2.021	2.423	2.704	2.971	3.307
60	1.296	1.671	2.000	2.390	2.660	2.915	3.232
120	1.289	1.658	1.980	2.358	2.617	2.860	3.160
∞	1.282	1.645	1.960	2.326	2.576	2.807	3.090

Table 1.10 Percentage points for the F -distribution $F_{0.05, \nu_2/\nu_1}$

ν_2/ν_1	1	2	3	4	5	6	7	8	9	10
1	161.40	199.50	215.70	224.60	230.20	234.00	236.80	238.90	240.50	241.90
2	18.51	19.00	19.16	19.25	19.30	19.33	19.35	19.37	19.38	19.40
3	10.13	9.55	9.28	9.12	9.01	8.94	8.89	8.85	8.81	8.79
4	7.71	6.94	6.59	6.39	6.26	6.16	6.09	6.04	6.00	5.96
5	6.61	5.79	5.41	5.19	5.05	4.95	4.88	4.82	4.77	4.74
6	5.99	5.14	4.76	4.53	4.39	4.28	4.21	4.15	4.10	4.06
7	5.59	4.74	4.35	4.12	3.97	3.87	3.79	3.73	3.68	3.64
8	5.32	4.46	4.07	3.84	3.69	3.58	3.50	3.44	3.39	3.35
9	5.12	4.26	3.86	3.63	3.48	3.37	3.29	3.23	3.18	3.14
10	4.95	4.10	3.71	3.48	3.33	3.22	3.14	3.07	3.02	2.98
11	4.84	3.98	3.59	3.36	3.20	3.09	3.01	2.95	2.90	2.85
12	4.75	3.89	3.49	3.26	3.11	3.00	2.91	2.85	2.80	2.75
13	4.67	3.81	3.41	3.18	3.03	2.92	2.83	2.77	2.71	2.67
14	4.60	3.74	3.34	3.11	2.96	2.85	2.76	2.70	2.65	2.60
15	4.54	3.68	3.29	3.06	2.90	2.79	2.71	2.64	2.59	2.54
16	4.49	3.63	3.24	3.01	2.85	2.74	2.66	2.59	2.54	2.49
17	4.45	3.59	3.20	2.96	2.81	2.70	2.61	2.55	2.49	2.45
18	4.41	3.55	3.16	2.93	2.77	2.66	2.58	2.51	2.46	2.41
19	4.38	3.52	3.13	2.90	2.74	2.63	2.54	2.48	2.42	2.38
20	4.35	3.49	3.10	2.87	2.71	2.60	2.51	2.45	2.39	2.35
21	4.32	3.47	3.07	2.84	2.68	2.57	2.49	2.42	2.37	2.32
22	4.30	3.44	3.05	2.82	2.66	2.55	2.46	2.40	2.34	2.30
23	4.28	3.42	3.03	2.80	2.64	2.53	2.44	2.37	2.32	2.27
24	4.26	3.40	3.01	2.78	2.62	2.51	2.42	2.36	2.30	2.25
25	4.24	3.39	2.99	2.76	2.60	2.49	2.40	2.34	2.28	2.24
26	4.23	3.37	2.98	2.74	2.59	2.47	2.39	2.32	2.27	2.22
27	4.21	3.35	2.96	2.73	2.57	2.46	2.37	2.31	2.25	2.20
28	4.20	3.34	2.95	2.71	2.56	2.45	2.36	2.29	2.24	2.19
29	4.18	3.33	2.93	2.70	2.55	2.43	2.35	2.28	2.22	2.18
30	4.17	3.32	2.92	2.69	2.53	2.42	2.33	2.27	2.21	2.16
40	4.08	3.23	2.84	2.61	2.45	2.34	2.25	2.18	2.12	2.08
60	4.00	3.15	2.76	2.53	2.37	2.25	2.17	2.10	2.04	1.99
120	3.92	3.07	2.68	2.45	2.29	2.17	2.09	2.02	1.96	1.91
∞	3.84	3.00	2.60	2.37	2.21	2.10	2.01	1.94	1.88	1.83

Table 1.11 Percentage points for the χ^2 distribution

v/χ^2_α	$\chi^2_{0.99}$	$\chi^2_{0.975}$	$\chi^2_{0.95}$	$\chi^2_{0.90}$	$\chi^2_{0.10}$	$\chi^2_{0.05}$	$\chi^2_{0.025}$	$\chi^2_{0.01}$
1	0	0.00	0.00	0.02	2.71	3.84	5.02	6.64
2	0.02	0.05	0.10	0.21	4.61	5.99	7.38	9.21
3	0.12	0.22	0.35	0.58	6.25	7.82	9.35	11.35
4	0.30	0.48	0.71	1.06	7.78	9.49	11.14	13.28
5	0.55	0.83	1.15	1.61	9.24	11.07	12.83	15.09
6	0.87	1.24	1.64	2.20	10.65	12.59	14.45	16.81
7	1.24	1.69	2.17	2.83	12.02	14.07	16.01	18.48
8	1.65	2.18	2.73	3.49	13.36	15.51	17.54	20.09
9	2.09	2.70	3.33	4.17	14.68	16.92	19.02	21.67
10	2.56	3.25	3.94	4.87	15.99	18.31	20.48	23.21
11	3.05	3.82	4.58	5.58	17.28	19.68	21.92	24.73
12	3.57	4.40	5.23	6.30	18.55	21.92	23.34	26.22
13	4.11	5.01	5.89	7.04	19.81	22.36	24.74	27.69
14	4.66	5.63	6.57	7.79	21.06	23.69	26.12	29.14
15	5.23	6.26	7.26	8.57	22.31	25.00	27.49	30.58
16	5.81	6.91	7.96	9.31	23.54	26.30	28.85	32.00
17	6.41	7.56	8.67	10.09	24.77	27.59	30.19	33.41
18	7.02	8.23	9.39	10.87	25.99	28.87	31.53	34.81
19	7.63	8.91	10.12	11.65	27.20	30.14	32.85	36.19
20	8.26	9.59	10.85	12.44	28.41	31.41	34.17	37.57
21	8.90	10.28	11.59	13.24	29.62	32.67	35.48	38.93
22	9.54	10.98	12.34	14.04	30.81	33.92	36.78	40.29
23	10.20	11.69	13.09	14.85	32.01	35.17	38.08	41.64
24	10.86	12.40	13.85	15.66	33.20	36.42	39.36	42.98
25	11.52	13.12	14.61	16.47	34.38	37.65	40.65	44.31
26	12.20	13.84	15.38	17.29	35.56	38.89	41.92	45.64
27	12.88	14.57	16.15	18.11	36.74	40.11	43.19	46.96
28	13.57	15.31	16.93	18.94	37.92	41.34	44.46	48.28
29	14.26	16.05	17.71	19.77	39.09	42.56	45.72	49.59
30	14.95	16.79	18.49	20.60	40.26	43.77	46.98	50.89
35	18.48	20.56	22.46	24.81	46.03	49.80	53.21	57.36
40	22.14	24.42	26.51	29.07	51.78	55.76	59.35	63.71
50	29.69	32.35	34.76	37.71	63.14	67.50	71.42	76.17
60	37.47	40.47	43.19	46.48	74.37	79.08	83.30	88.39
70	45.43	48.75	51.74	55.35	85.50	90.53	95.03	100.44
80	53.53	57.15	60.39	64.30	96.55	101.88	106.63	112.34
90	61.74	65.64	69.12	73.31	107.54	113.15	118.14	124.13
100	70.05	74.22	77.93	82.38	118.47	124.34	129.57	135.81
110	78.45	82.86	86.79	91.50	129.36	135.48	140.92	147.42
120	86.91	91.57	95.70	100.65	140.20	146.57	152.22	158.96

References

1. Pham, H.: Software Reliability. Springer, Berlin, Heidelberg (2000)
2. Weibull, W.: A statistical distribution function of wide applicability. *J. Appl. Mech.* **18**, 293–297 (1951)
3. Pham, H.: A Vtub-shaped hazard rate function with applications to system safety. *Int. J. Reliab. Appl.* **3**(1), 1–16 (2002)
4. Chen, Z.: Exact confidence interval for the shape parameter of a loglogistic distribution. *J. Stat. Comput. Sim.* **56**, 193–211 (1997)
5. Nelson, W.: Applied Life Data Analysis. Wiley, New York (1982)
6. Zhao, M.: Statistical reliability change-point estimation models. In: Pham, H. (ed.) *Handbook of Reliability Engineering*, pp. 157–163. Springer, Berlin, Heidelberg (2003)
7. Jelinski, Z., Moranda, P.B.: Software reliability research. In: Freiberger, W. (ed.) *Statistical Computer Performance Evaluation*. Academic, New York (1972)
8. Musa, J.D., Lannino, A., Okumoto, K.: *Software Reliability: Measurement, Prediction, and Application*. McGraw-Hill, New York (1987)
9. Feller, W.: *An Introduction to Probability Theory and Its Applications*, 3rd edn. Wiley, New York (1994)
10. Wang, H., Pham, H.: A quasi renewal process and its applications in imperfect maintenance. *Int. J. Syst. Sci.* **27**(10), 1055–1062 (1996)
11. Devore, J.L.: *Probability and Statistics for Engineering and the Sciences*, 3rd edn. Brooks Cole, Pacific Grove (1991)
12. Gnedenko, B.V., Ushakov, I.A.: *Probabilistic Reliability Engineering*. Wiley, New York (1995)
13. Hahn, J.G., Meeker, W.Q.: *Statistical Intervals: a Guide for Practitioners*. Wiley, New York (1991)
14. Pham, H.: A distribution function and its applications in software reliability. *Int. J. Perform. Eng.* **15**(5), 1306–1313 (2019)
15. Cordeiro, G.M., Silva, G.O., Ortega, E.M.: An extended-G geometric family. *J. Stat. Distrib. Appl.* **3**(3), 1–16 (2016)



Hoang Pham is a distinguished professor and former chairman of the Department of Industrial and Systems Engineering at Rutgers University. He is the author or coauthor of 7 books and has published over 200 journal articles, 100 conference papers, and edited 20 books. His numerous awards include the 2009 IEEE Reliability Society *Engineer of the Year Award*. He is a fellow of the IEEE and IISE.



Generalized Statistical Distributions

2

Gauss M. Cordeiro and Artur Lemonte

Contents

2.1	Beta-G Family	43
2.2	Kumaraswamy-G Family	44
2.3	Gamma-G Family	45
2.4	Marshall and Olkin-G Family	46
2.5	Linear Representations	46
2.6	Mathematical Properties	47
2.6.1	Moments	47
2.6.2	Other Measures	48
2.6.3	Generating Function	49
2.7	T-X Family	49
2.8	Fitting Distributions	51
2.9	Real Data Illustrations	51
	References	52

Abstract

There has been an increased interest in developing generalized families of distributions by introducing additional shape parameters to a baseline cumulative distribution. This mechanism has proved to be useful to make the generated distributions more flexible especially for studying tail properties than existing distributions and for improving their goodness-of-fit statistics to the data under study. Let $G(x)$ be the cumulative distribution function (CDF) of a baseline distribution and $g(x) = dG(x)/dx$ be the associated probability density function (PDF). We present generalized families with one and two additional shape parameters by transforming the CDF $G(x)$ according to four

important generators. These families are important for modeling data in several engineering areas. Many special distributions in these families are discussed by Tahir and Nadarajah (An Acad Bras Cienc 87(2):539–568, 2015).

Keywords

Distribution theory · Extended distributions · Failure rate function · Repaired systems · Reliability studies · Survival analysis

2.1 Beta-G Family

Eugene et al. [6] defined the CDF of the beta-G family with two additional shape parameters $a > 0$ and $b > 0$ by

$$F_{\text{beta-G}}(x) = I_{G(x)}(a, b) = \frac{1}{B(a, b)} \int_0^{G(x)} w^{a-1} (1-w)^{b-1} dw, \quad x \in \mathbb{R}, \quad (2.1)$$

where $B(a, b) = \Gamma(a + b)/[\Gamma(a)\Gamma(b)]$ is the beta function, $\Gamma(\alpha) = \int_0^\infty w^{\alpha-1} e^{-w} dw$ is the gamma function, and $I_y(a, b) = B(a, b)^{-1} \int_0^y w^{a-1} (1-w)^{b-1} dw$ is the incomplete beta function ratio. The PDF corresponding to (2.1) has a very simple form

$$f_{\text{beta-G}}(x) = \frac{g(x)}{B(a, b)} G(x)^{a-1} \{1 - G(x)\}^{b-1}, \quad x \in \mathbb{R}. \quad (2.2)$$

The beta-G family has been receiving increased attention over the last years. More than 50 distributions were published in this family, the majority of them cited by Tahir and Nadarajah [22, Table 3].

The PDF $f_{\text{beta-G}}(x)$ will be most tractable when the functions $G(x)$ and $g(x)$ have simple analytic expressions. Except for some special choices for $G(x)$ in (2.1), the PDF (2.2) will be difficult to deal with in generality. If $g(x)$ is a symmetric distribution around zero, then $f_{\text{beta-G}}(x)$ will also be a

G. M. Cordeiro (✉)
UFPE, Recife, Brazil
e-mail: gauss@de.ufpe.br

A. Lemonte
UFRN, Natal, Brazil

symmetric distribution when $a = b$. We can generate observations from the beta-G family using beta random variables and the quantile function (QF) of the baseline G distribution, say, $Q_G(u) = G^{-1}(u)$. In fact, if V has a beta distribution with parameters $a > 0$ and $b > 0$, then $X = Q_G(V)$ follows the PDF (2.2).

The CDF of the beta-Weibull (beta-W) distribution (with four positive parameters) is defined by taking the Weibull CDF $G_{\lambda,c}(x) = 1 - \exp\{-(\lambda x)^c\}$ with shape parameter $c > 0$ and scale parameter $\lambda > 0$ in Eq. (2.1). The PDF and the hazard rate function (HRF) of the beta-W distribution are (using $I_x(a, b) = I_{1-x}(b, a)$)

$$f_{\text{beta-W}}(x) = \frac{c\lambda^c}{B(a, b)} x^{c-1} \exp\{-b(\lambda x)^c\} \times [1 - \exp\{-(\lambda x)^c\}]^{a-1}, \quad x > 0,$$

and

$$\tau_{\text{beta-W}}(x) = \frac{c\lambda^c x^{c-1} \exp\{-b(\lambda x)^c\} [1 - \exp\{-(\lambda x)^c\}]^{a-1}}{B(a, b) I_{\exp\{-(\lambda x)^c\}}(b, a)}, \quad x > 0,$$

respectively.

Birnbaum and Saunders [3] introduced a probability distribution which is commonly used in reliability studies. The CDF of the beta-Birnbaum-Saunders (beta-BS) distribution is given by $F(x) = I_{\Phi(v)}(a, b)$, for $x > 0$, where $v = v(x; \alpha, \beta) = \alpha^{-1} \rho(x/\beta)$, $\rho(z) = z^{1/2} - z^{-1/2}$, $\Phi(\cdot)$ is the standard normal CDF, and $\alpha > 0$ and $\beta > 0$. The corresponding PDF is

$$f_{\text{beta-BS}}(x) = \frac{\kappa(\alpha, \beta) x^{-3/2} (x + \beta) \exp\{-\tau(x/\beta)/(2\alpha^2)\}}{B(a, b) \Phi(v)^{1-a} [1 - \Phi(v)]^{1-b}}, \quad x > 0,$$

where $\kappa(\alpha, \beta) = \exp(\alpha^{-2})/(2\alpha\sqrt{2\pi\beta})$ and $\tau(z) = z + z^{-1}$. For $a = b = 1$, the beta-BS becomes equal to the BS distribution.

Finally, another example is the beta-log-logistic (beta-LL) distribution (see, e.g., Lemonte [9]), whose PDF with positive parameters a, b, α , and β is given by

$$f(x) = \frac{(\beta/\alpha)}{B(a, b)} \frac{(x/\alpha)^{a\beta-1}}{[1 + (x/\alpha)^\beta]^{a+b}}, \quad x > 0. \quad (2.3)$$

Evidently, the beta-LL density function does not involve any complicated function and can be easily computed from Eq. (2.3). The beta-LL distribution can be applied in survival analysis, hydrology, and economics, among others, as the LLog distribution and can be used to model reliability

problems. The beta-LL distribution allows for greater flexibility of its tails and can be widely applied in many areas.

The study of the PDF (2.3) is important since it also includes some special sub-models. The log-logistic (LLog) distribution arises as the basic exemplar when $a = b = 1$. The exponentiated LLog (ELLog) distribution corresponds to $b = 1$. Other special sub-model arises for $a = 1$ as the new Lehmann type II LLog (LeLLog) distribution. For a and b positive integers, the beta-LL density function becomes the density function of the a th order statistic from the LLog distribution in a sample of size $a + b - 1$. However, Eq. (2.3) can also alternatively be extended, when a and b are real non-integers, to define fractional LLog order statistics distributions.

2.2 Kumaraswamy-G Family

Cordeiro and de Castro [5] introduced the Kumaraswamy-G (“Kw-G” for short) family with CDF and PDF

$$F_{\text{Kw-G}}(x) = 1 - \{1 - G(x)^a\}^b, \quad x \in \mathbb{R}, \quad (2.4)$$

and

$$f_{\text{Kw-G}}(x) = a b g(x) G(x)^{a-1} \{1 - G(x)^a\}^{b-1}, \quad x \in \mathbb{R}, \quad (2.5)$$

respectively. The Kw-G family has the same parameters of the baseline G distribution plus two extra shape parameters $a > 0$ and $b > 0$. Each new Kw-G distribution can be constructed from a specified G distribution. For $a = b = 1$, the G distribution is a special model of the Kw-G family with a continuous crossover toward cases with different shapes (e.g., a particular combination of skewness and kurtosis). One major benefit of the Kw-G PDF (2.5) is its ability of fitting skewed data that cannot be properly fitted by existing distributions. Further, this family allows for greater flexibility of its tails and can be widely applied in many areas of engineering. Tahir and Nadarajah [22, Table 5] described almost 25 published distributions belonging to the Kw-G family.

The simulation from the Kw-G distribution is very easy. The QF corresponding to (2.4) is directly obtained from the QF associated with $G(x)$ by

$$Q_{\text{Kw-G}}(u) = F_{\text{Kw-G}}^{-1}(u) = Q_G \left\{ \left[1 - (1 - u)^{1/b} \right]^{1/a} \right\}, \quad u \in (0, 1). \quad (2.6)$$

Then, we can generate Kw-G variates by

$$X = Q_G \left\{ \left[1 - (1 - U)^{1/b} \right]^{1/a} \right\},$$

where U is a uniform variate on the unit interval $(0, 1)$.

We provide two examples of the Kw-G distributions. First, the Kw-Weibull PDF (for $x > 0$) follows from the Weibull CDF (with parameters $\lambda > 0$ and $c > 0$) introduced in Sect. 2.1. It has the form

$$f_{\text{Kw-W}}(x) = \frac{abc\beta^c x^{c-1} \exp\{-(\lambda x)^c\} [1 - \exp\{-(\lambda x)^c\}]^{a-1}}{1 - [1 - \exp\{-(\lambda x)^c\}]^a}^{1-b}, \quad x > 0.$$

For $c = 1$, we obtain as a special model the Kw-exponential distribution. The exponentiated Weibull (EW), which is well suited for modeling bathtub failure rate lifetime data, is also a special case when $b = 1$.

Second, the Gumbel distribution is frequently used to model the distribution of the maximum (or the minimum) of a number of samples of various distributions. The Kw-Gumbel density function is defined from the Gumbel CDF $G(x) = 1 - \exp\{-\exp(-\frac{x-\mu}{\sigma})\}$ (for $x \in \mathbb{R}$) with location parameter $\mu > 0$ and scale parameter $\sigma > 0$, whose mean and variance are equal to $\mu - \gamma\sigma$ and $\pi^2\sigma^2/6$, respectively, and γ is Euler's constant ($\gamma \approx 0.57722$). It is given by

$$f_{\text{Kw-Gu}}(x) = \frac{ab \exp\{z - \exp(z)\} [1 - \exp\{-\exp(-z)\}]^{a-1}}{\sigma \{1 - [1 - \exp\{-\exp(-z)\}]^a\}^{1-b}},$$

where $z = (y - \mu)/\sigma$.

2.3 Gamma-G Family

Zografos and Balakrishnan [23] and Ristić and Balakrishnan [20] defined the gamma-G (Γ -G) family with an extra shape parameter $a > 0$ by the CDF (for $x \in \mathbb{R}$)

$$F_{\Gamma\text{-G}}(x) = \frac{\gamma(a, -\log[1 - G(x)])}{\Gamma(a)} = \frac{1}{\Gamma(a)} \int_0^{-\log[1 - G(x)]} t^{a-1} e^{-t} dt,$$

where $\gamma(a, z) = \int_0^z t^{a-1} e^{-t} dt$ is the incomplete gamma function. The PDF of the Γ -G family has the form

$$f_{\Gamma\text{-G}}(x) = \frac{1}{\Gamma(a)} \{-\log[1 - G(x)]\}^{a-1} g(x), \quad x \in \mathbb{R}. \quad (2.7)$$

Each new Γ -G distribution can be determined from a given baseline distribution. For $a = 1$, the G distribution is a basic exemplar of the Γ -G family.

Zografos and Balakrishnan [23] and Ristić and Balakrishnan [20] presented a physical motivation for the Γ -G family: if $X_{L(1)}, \dots, X_{L(n)}$ are lower record values from a sequence of independent random variables with common PDF $g(\cdot)$, then the PDF of the n th lower record value has the form (2.7). If Z is a gamma random variable with unit scale parameter and shape parameter $a > 0$, then $X = Q_G(1 - e^{-Z})$ has density (2.7). So, the Γ -G distribution is easily generated from the gamma distribution and the QF of G.

Let $G_{\lambda,c}(x)$ be the Weibull CDF (with scale parameter $\lambda > 0$ and shape parameter $c > 0$) given in Sect. 2.1. The Γ -Weibull (Γ -W) density function (for $x > 0$) can be expressed as

$$f_{\Gamma\text{-W}}(x) = \frac{c\lambda^{ca}}{\Gamma(a)} x^{ac-1} \exp\{-(\lambda x)^c\}. \quad (2.8)$$

Equation (2.8) extends some distributions previously discussed in the literature. In fact, it is identical to the generalized gamma distribution [21]. The Weibull distribution is a basic exemplar when $a = 1$, whereas the gamma distribution follows when $c = 1$. The half-normal distribution is obtained for $a = 3$ and $c = 2$. In addition, the log-normal distribution is a limiting special case when a tends to infinity.

The Γ -normal distribution is defined from (2.7) by taking $G(x)$ and $g(x)$ to be the CDF and PDF of the normal $N(\mu, \sigma^2)$ distribution. Its density function (for $x \in \mathbb{R}$) has the form

$$f_{\Gamma\text{-N}}(x) = \frac{1}{\Gamma(a)} \left\{ -\log \left[1 - \Phi\left(\frac{x-\mu}{\sigma}\right) \right] \right\}^{a-1} \phi\left(\frac{x-\mu}{\sigma}\right), \quad (2.9)$$

where $\mu \in \mathbb{R}$ is a location parameter, $\sigma > 0$ is a scale parameter, and $\phi(\cdot)$ is the standard normal density. For $\mu = 0$ and $\sigma = 1$, we obtain the standard Γ -normal distribution. Further, it coincides with the normal distribution when $a = 1$.

Consider the Gumbel distribution with location parameter $\mu \in \mathbb{R}$ and scale parameter $\sigma > 0$ as defined in Sect. 2.2. The Gumbel PDF is

$$g(x) = \frac{1}{\sigma} \exp \left\{ \left(\frac{x-\mu}{\sigma} \right) - \exp \left(\frac{x-\mu}{\sigma} \right) \right\}, \quad x \in \mathbb{R},$$

where $\mu \in \mathbb{R}$ and $\sigma > 0$. The Γ -Gumbel (Γ -Gu) PDF can be expressed as

$$f_{\Gamma\text{-Gu}}(x) = \frac{1}{\sigma\Gamma(a)} \exp \left\{ (a-1) \left(\frac{x-\mu}{\sigma} \right) + \left(\frac{x-\mu}{\sigma} \right) - \exp \left(\frac{x-\mu}{\sigma} \right) \right\}.$$

Clearly, the Gumbel distribution corresponds to $a = 1$.

2.4 Marshall and Olkin-G Family

Marshall and Olkin [11] pioneered a general method to expand a distribution by adding an extra shape parameter. The CDF of their family (for $\theta > 0$) is

$$\begin{aligned} F_{\text{MO-G}}(x) &= \frac{G(x)}{\theta + (1 - \theta)G(x)} \\ &= \frac{G(x)}{1 - (1 - \theta)[1 - G(x)]}, \quad x \in \mathbb{R}. \end{aligned} \quad (2.10)$$

The density function corresponding to (2.10) is

$$f_{\text{MO-G}}(x) = \frac{\theta g(x)}{[\theta + (1 - \theta)G(x)]^2}, \quad x \in \mathbb{R}. \quad (2.11)$$

For $\theta = 1$, $f_{\text{MO-G}}(x)$ is equal to $g(x)$, and, for different values of θ , $f_{\text{MO-G}}(x)$ can be more flexible than $g(x)$. The extra parameter θ is called ‘‘tilt parameter,’’ since the HRF of the MO-G family is shifted below ($\theta > 1$) or above ($0 < \theta < 1$) of the baseline HRF. Equation (2.11) provides a useful mechanism to generate new distributions from existing ones. The advantage of this approach for constructing new distributions lies in its flexibility to model both monotonic and non-monotonic HRFs even when the baseline HRF may be monotonic. Tahir and Nadarajah [22, Table 2] presented 30 distributions belonging to the MO-G family. Further, this family is easily generated from the baseline QF by $Q_{\text{MO-G}}(u) = Q_G(\theta u / [\theta u + 1 - u])$ for $u \in (0, 1)$.

Marshall and Olkin considered the exponential and Weibull distributions for the baseline G and derived some structural properties of the generated distributions. The special case that G is an exponential distribution refers to a two-parameter competitive model to the Weibull and gamma distributions. A simple interpretation of (2.10) can be given as follows. Let T_1, \dots, T_N be a sequence of independent and identically distributed (i.i.d.) random variables with survival function (SF) $\bar{G}(x) = 1 - G(x)$, and let N be a positive integer random variable independent of the T_i 's defined by the probability generating function (PGF) of a geometric distribution with parameter θ , say, $\tau(z; \theta) = \theta z [1 - (1 - \theta)z]^{-1}$. Then, the inverse of $\tau(z; \theta)$ becomes $\tau^{-1}(z; \theta) = \tau(z; \theta^{-1})$. We can verify that Eq. (2.10) comes from $1 - F_{\text{MO-G}}(x) = \tau(\bar{G}(x); \theta)$ for $0 < \theta < 1$ and $1 - F_{\text{MO-G}}(x) = \tau(\bar{G}(x); \theta^{-1})$ for $\theta > 1$. For both cases, $1 - F_{\text{MO-G}}(x)$ represents the SF of $\min\{T_1, \dots, T_N\}$, where N has PGF $\tau(z; \cdot)$ with probability parameters θ or θ^{-1} .

Nadarajah and Haghghi [14] introduced a two-parameter generalization of the exponential distribution as an alternative to the gamma and Weibull distributions with CDF (NH stands for the authors) $G_{\text{NH}}(x) = 1 - \exp[1 - (1 + \lambda x)^\alpha]$, $x \geq 0$, where $\lambda > 0$ is the scale parameter and $\alpha > 0$ is the shape parameter. The associated density is $g_{\text{NH}}(x) = \alpha \lambda (1 +$

$\lambda x)^{\alpha-1} \exp[1 - (1 + \lambda x)^\alpha]$, $x \geq 0$. The function $g_{\text{NH}}(x)$ is always monotonically decreasing with $g_{\text{NH}}(0) = \alpha \lambda$. These authors pointed out that $g_{\text{NH}}(x)$ has the attractive feature of always having the zero mode. They also showed that larger values of α will lead to faster decay of the upper tails. The SF and HRF are $S_{\text{NH}}(x) = \exp[1 - (1 + \lambda x)^\alpha]$ and $\tau_{\text{NH}}(x) = \alpha \lambda (1 + \lambda x)^{\alpha-1}$, respectively. Like the Weibull distribution, the SF and HRF of the NH distribution have closed-form expressions. Additionally, the HRF can be monotonically increasing for $\alpha > 1$ and monotonically decreasing for $\alpha < 1$. For $\alpha = 1$, the HRF is constant. So, the major weakness of the NH distribution is its inability to accommodate non-monotone HRFs (i.e., bathtub and unimodal shapes).

The MO-NH distribution was defined by Lemonte et al. [10] by combining the MO generator and the NH distribution. Its CDF is

$$F_{\text{MO-NH}}(x) = \frac{1 - \exp[1 - (1 + \lambda x)^\alpha]}{1 - (1 - \theta) \exp[1 - (1 + \lambda x)^\alpha]}, \quad x \geq 0, \quad (2.12)$$

where the parameters $\alpha > 0$ and $\beta > 0$ control the shapes of the distribution and the parameter $\lambda > 0$ is the scale parameter. Clearly, the MO-NH distribution is equal to the NH distribution when $\theta = 1$. For $\alpha = 1$, we obtain the MO-exponential (MO-E) distribution. The exponential distribution follows as a special model when $\alpha = \theta = 1$.

The MO-NH PDF follows by differentiating (2.12) as

$$f_{\text{MO-NH}}(x) = \alpha \theta \lambda \frac{(1 + \lambda x)^{\alpha-1} \exp[1 - (1 + \lambda x)^\alpha]}{[1 - (1 - \theta) \exp[1 - (1 + \lambda x)^\alpha]]^2}, \quad x \geq 0. \quad (2.13)$$

Equation (2.13) can take various forms depending on the values of the shape parameters α and θ . In particular, it can be a decreasing function, or it can be a skewed unimodal density. Additionally, the MO-NH density function can present some other interesting forms. In fact, this distribution is very versatile since the additional parameter θ has substantial effects on its skewness and kurtosis.

The MO-NH HRF can be constant, decreasing, increasing, unimodal, and bathtub-shaped. Further, it can also be decreasing-increasing-decreasing. The MO-NH distribution is a good alternative to many existing lifetime distributions in modeling real data due to the great flexibility of its HRF.

2.5 Linear Representations

Linear representations for the PDFs of the four families defined previously can be derived using the concept of exponentiated distributions. For an arbitrary baseline CDF $G(x)$, the exponentiated-G (exp-G) distribution with parameter $a > 0$ has CDF and PDF in the forms

$$\Pi_a(x) = G(x)^a \quad \text{and} \quad \pi_a(x) = a g(x) G(x)^{a-1},$$

respectively. The properties of exponentiated distributions have been studied by many authors in recent years; see [12] for exponentiated Weibull, [8] for exponentiated Pareto, [7] for exponentiated exponential, and [13] for exponentiated gamma distribution. Tahir and Nadarajah [22] cited almost 30s exponentiated distributions in their Table 2.1.

Linear representations for the four PDFs in terms of exp-G densities are important to determine their mathematical properties from those of the exp-G distributions. They can follow from the papers described below. For the beta-G density (2.2), [16] demonstrated that

$$f_{\text{beta-G}}(x) = \sum_{i=0}^{\infty} w_i^{\text{beta-G}} \pi_{a+i}(x),$$

where

$$w_i^{\text{beta-G}} = w_i^{\text{beta-G}}(a, b) = \frac{(-1)^i}{(a+i)B(a, b)} \binom{b-1}{i}.$$

Nadarajah et al. [17] determined a linear combination for the Kw-G density (2.5), namely,

$$f_{\text{Kw-G}}(x) = \sum_{i=0}^{\infty} w_i^{\text{Kw-G}} \pi_{(i+1)a}(x), \quad (2.14)$$

where

$$w_i^{\text{Kw-G}} = w_i^{\text{Kw-G}}(b) = \frac{(-1)^i b}{(i+1)} \binom{b-1}{i}.$$

The linear combination for the Γ -G density (2.7) was derived by Castellares and Lemonte [4] as

$$f_{\Gamma\text{-G}}(x) = \sum_{i=0}^{\infty} w_i^{\Gamma\text{-G}} \pi_{a+i}(x),$$

where

$$w_i^{\Gamma\text{-G}} = w_i^{\Gamma\text{-G}}(a) = \frac{\varphi_i(a)}{(a+i)},$$

$$\varphi_0(a) = \frac{1}{\Gamma(a)}, \quad \varphi_i(a) = \frac{(a-1)}{\Gamma(a)} \psi_{i-1}(i+a-2), \quad i \geq 1,$$

and $\psi_{i-1}(\cdot)$ are the Stirling polynomials defined by

$$\begin{aligned} \psi_{n-1}(x) &= \frac{(-1)^{n-1}}{(n+1)!} \left[H_n^{n-1} - \frac{x+2}{n+2} H_n^{n-2} + \frac{(x+2)(x+3)}{(n+2)(n+3)} H_n^{n-3} \right. \\ &\quad \left. - \dots + (-1)^{n-1} \frac{(x+2)(x+3) \dots (x+n)}{(n+2)(n+3) \dots (2n)} H_n^0 \right], \end{aligned}$$

where H_n^m are positive integers given recursively by $H_{n+1}^m = (2n+1-m)H_n^m + (n-m+1)H_n^{m-1}$, with $H_0^0 = 1, H_{n+1}^0 = 1 \times 3 \times 5 \times \dots \times (2n+1), H_{n+1}^n = 1$.

Following Barreto-Souza et al. [2], the MO-G density (2.11) admits the linear combination

$$f_{\text{MO-G}}(x) = \sum_{i=0}^{\infty} w_i^{\text{MO-G}} \pi_{i+1}(x),$$

where the coefficients are (for $i = 0, 1, \dots$)

$$\begin{aligned} w_i^{\text{MO-G}} &= w_i^{\text{MO-G}}(\theta) \\ &= \begin{cases} \frac{(-1)^i \theta}{(i+1)} \sum_{j=i}^{\infty} \binom{j}{i} (j+1) \bar{\theta}^j, & \theta \in (0, 1), \\ \theta^{-1} (1-\theta^{-1})^i, & \theta > 1, \end{cases} \end{aligned}$$

and $\bar{\theta} = 1 - \theta$.

2.6 Mathematical Properties

In this section, we present only some mathematical properties for the Kw-G family from Eq. (2.14). The properties for the beta-G, Γ -G, and MO-G families can be obtained from their linear representations in a similar manner. Henceforth, let $X \sim \text{Kw-G}(a, b)$ and $Y_i \sim \text{exp-G}((i+1)a)$ for $i \geq 0$.

2.6.1 Moments

Nadarajah et al. [17] derived explicit expressions for the moments of X as linear functions of probability weighted moments (PWMs) of the baseline G. A first representation for the n th moment of X follows from (2.14) as

$$\mu'_n = E(X^n) = \sum_{i=0}^{\infty} w_i^{\text{Kw-G}} E(Y_i^n). \quad (2.15)$$

Equation (2.15) holds for $b > 0$ real non-integer. If b is a positive integer, the index i in this expansion stops at $b - 1$. Explicit expressions for moments of several exponentiated distributions are reported by Nadarajah and Kotz [15] which can be used to calculate μ'_n .

A second representation for μ'_n can be determined from (2.14) as

$$\mu'_n = a \sum_{i=0}^{\infty} (i+1) w_i^{\text{Kw-G}} \tau_n((i+1)a-1), \quad (2.16)$$

where the integral $\tau_n(a) = \int_{-\infty}^{\infty} y^n G(y)^a g(y) dy$ can be expressed in terms of the baseline QF as

$$\tau_n(a) = \int_0^1 Q_G(u)^n u^a du. \quad (2.17)$$

The central moments (μ_n) and cumulants (κ_n) of X can be calculated recursively as

$$\mu_n = \sum_{k=0}^n (-1)^k \binom{n}{k} \mu_1^k \mu'_{n-k} \quad \text{and}$$

$$\kappa_n = \mu'_n - \sum_{k=1}^{n-1} \binom{n-1}{k-1} \kappa_k \mu'_{n-k},$$

respectively, where $\kappa_1 = \mu'_1$. The skewness $\gamma_1 = \kappa_3/\kappa_2^{3/2}$ and kurtosis $\gamma_2 = \kappa_4/\kappa_2^2$ are obtained from the second, third, and fourth cumulants.

The ordinary moments of several Kw-G distributions can be determined directly from (2.16) and (2.17). For example, the moments of the Kw-exponential (with parameter $\lambda > 0$) and Kw-Pareto, where $G(x) = 1 - (1+x)^{-\nu}$ and $\nu > 0$, are given by

$$\mu'_n = a n! \lambda^n \sum_{i,j=0}^{\infty} \frac{(-1)^{n+j} (i+1) \binom{(i+1)a-1}{j} w_i^{\text{Kw-G}}}{(j+1)^{n+1}},$$

and

$$\mu'_n = a \sum_{i,j=0}^{\infty} (-1)^{n+j} (i+1) w_i^{\text{Kw-G}} \binom{n}{j} B((i+1)a, 1 - j\nu^{-1}),$$

respectively. For the Kw-standard logistic, where $G(x) = \{1 + \exp(-x)\}^{-1}$, we obtain

$$\mu'_n = a \sum_{i=0}^{\infty} (i+1) w_i^{\text{Kw-G}} \left(\frac{\partial}{\partial t} \right)^n B(t + (i+1)a, 1-t) \Big|_{t=0}.$$

The shape of many distributions can be usefully described by what we call the incomplete moments. These types of moments, and more importantly the first moment, play the main role for measuring inequality such as the mean deviations, income quantiles, and Lorenz and Bonferroni curves. The n th incomplete moment of X , say, $m_n(y) = E(X^n | X < w)$, can be expressed as

$$m_n(y) = a \sum_{i=0}^{\infty} (i+1) w_i^{\text{Kw-G}} \int_0^{G(w)} Q_G(u)^n u^{(i+1)a-1} du. \quad (2.18)$$

Equation (2.18) can be computed (at least numerically) for most baseline G distributions.

2.6.2 Other Measures

The mean deviations of X about the mean ($\delta_1(X)$) and about the median ($\delta_2(X)$) are given by

$$\delta_1(X) = E(|X - \mu'_1|) = 2\mu'_1 F_{\text{Kw-G}}(\mu'_1) - 2m_1(\mu'_1),$$

and

$$\delta_2(X) = E(|X - M|) = \mu'_1 - 2m_1(M),$$

respectively. Here, $\mu'_1 = E(X)$, $M = Q_G\{[1 - 2^{-1/b}]^{1/a}\}$ is the median, $F_{\text{Kw-G}}(\mu'_1)$ comes from (2.4), and $m_1(z)$ (the first incomplete moment) follows from (2.18) as

$$m_1(z) = a \sum_{i=0}^{\infty} (i+1) w_i^{\text{Kw-G}} T_i(z),$$

where

$$T_i(z) = \int_0^{G(z)} Q_G(u) u^{(i+1)a-1} du.$$

The mean deviations of any Kw-G distribution can be computed using the last equation and the baseline QF. Taking the generalized binomial expansion, the mean deviations of the Kw-exponential (with parameter λ), Kw-standard logistic, and Kw-Pareto (with parameter $\nu > 0$) are given by

$$T_i(z) = \lambda^{-1} \Gamma((i+1)a + 1) \sum_{j=0}^{\infty} \frac{(-1)^j \{1 - \exp(-j\lambda z)\}}{\Gamma((i+1)a + 1 - j)(j+1)!},$$

$$T_i(z) = \frac{1}{\Gamma(k)} \sum_{j=0}^{\infty} \frac{(-1)^j \Gamma((i+1)a + j) \{1 - \exp(-jz)\}}{(j+1)!},$$

and

$$T_i(z) = \sum_{j=0}^{\infty} \sum_{r=0}^j \frac{(-1)^j \binom{(i+1)a}{j} \binom{j}{r} z^{1-r\nu}}{(1-r\nu)},$$

respectively.

An alternative representation for $m_1(z)$ follows from (2.14) as

$$m_1(z) = \sum_{i=0}^{\infty} w_i^{\text{Kw-G}} J_i(z), \quad (2.19)$$

where

$$J_i(z) = \int_{-\infty}^z x \pi_{(i+1)a}(x) dx. \quad (2.20)$$

Equation (2.20) is the basic quantity to compute the mean deviations of the exp-G distributions. Hence, the Kw-G mean

deviations depend only on the mean deviations of the exp-G distribution. A simple application is provided by the Kw-Weibull distribution. The PDF (for $x > 0$) of the EW distribution with power parameter $(i + 1)a$ is

$$\pi_{(i+1)a}(x) = a(i + 1)c\lambda^c x^{c-1} \exp\{-(\lambda x)^c\} \times [1 - \exp\{-(\lambda x)^c\}]^{(i+1)a-1},$$

and then

$$J_i(z) = (i + 1)a c \lambda^c \sum_{r=0}^{\infty} (-1)^r \binom{(i + 1)a - 1}{r} \times \int_0^z x^c \exp\{-(r + 1)(\lambda x)^c\} dx.$$

We can write $J_i(z)$ in terms of the incomplete gamma function

$$J_i(z) = (i + 1)a \lambda^{-1} \sum_{r=0}^{\infty} \frac{(-1)^r \binom{(i+1)a-1}{r}}{(r + 1)^{1+c^{-1}}} \gamma(1 + c^{-1}, (r + 1)(\lambda z)^c).$$

Equations (2.19) and (2.20) can be used to find Bonferroni and Lorenz curves defined, for a given probability p , by $B(p) = m_1(q)/(p\mu'_1)$ and $L(p) = m_1(q)/\mu'_1$, respectively, where $\mu'_1 = E(X)$ and $q = Q_G\{[1 - (1 - p)^{1/b}]^{1/a}\}$.

2.6.3 Generating Function

We now provide three representations of the moment generating function (MGF) $M(t) = E(e^{tX})$ of X . The first one comes from (2.14) as

$$M(t) = a \sum_{i=0}^{\infty} (i + 1) w_i^{\text{Kw-G}} E[e^{tX} U^{(i+1)a+1}],$$

where U is a uniform random variable on the unit interval. Note that X and U are not independent.

A second representation for $M(t)$ can be written from (2.14) as

$$M(t) = \sum_{i=0}^{\infty} w_i^{\text{Kw-G}} M_i(t),$$

where $M_i(t)$ is the MGF of Y_i . Hence, for several Kw-G distributions, $M(t)$ can be determined from the exp-G generating function.

A third representation for $M(t)$ can be derived from (2.14) as

$$M(t) = a \sum_{i=0}^{\infty} (i + 1) w_i^{\text{Kw-G}} \rho(t, a(i + 1) - 1), \quad (2.21)$$

where the function $\rho(t, a) = \int_{-\infty}^{\infty} e^{tx} G(x)^a g(x) dx$ can be calculated from the baseline QF as

$$\rho(t, a) = \int_0^1 \exp\{tQ_G(u)\} u^a du. \quad (2.22)$$

We can obtain the MGFs of several Kw-G distributions from (2.21) and (2.22). For example, the MGFs of the Kw-exponential (with parameter λ), Kw-standard logistic, and Kw-Pareto (with parameter $\nu > 0$) are

$$M(t) = a \sum_{i=0}^{\infty} (i + 1) w_i^{\text{Kw-G}} B((i + 1)a, 1 - \lambda t),$$

$$M(t) = a \sum_{i=0}^{\infty} (i + 1) w_i^{\text{Kw-G}} B(t + (i + 1)a, 1 - t),$$

and

$$M(t) = a e^{-t} \sum_{i,r=0}^{\infty} \frac{(i + 1) w_i^{\text{Kw-G}} B((i + 1)a, 1 - r\nu^{-1})}{r!} t^r,$$

respectively.

2.7 T-X Family

Alzaatreh et al. [1] proposed the T-X family of distributions as follows. Let $r(t)$ be the PDF and $R(t)$ be the CDF of a random variable $(rv) T \in [a, b]$ for $-\infty < a < b < \infty$, and let $W[G(x)]$ be a function of the CDF $G(x)$ of some rv X defined on a standard probability space such that $W[G(x)]$ satisfies the following conditions:

- (i) $W[G(x)] \in [a, b]$.
- (ii) $W[G(x)]$ is differentiable and monotonically nondecreasing.
- (iii) $\lim_{x \rightarrow -\infty} W[G(x)] = a$ and $\lim_{x \rightarrow \infty} W[G(x)] = b$.

The T - X family is defined by

$$F(x) = \int_a^{W[G(x)]} r(t) dt = R(W[G(x)]), \quad (2.23)$$

where $W[G(x)]$ satisfies the conditions (i)–(iii). The PDF related to (2.23) turns out to be

$$f(x) = r(W[G(x)]) \frac{dW[G(x)]}{dx}. \quad (2.24)$$

Equation (2.23) can be used to define several families such as the beta-G, Kw-G, and Γ -G introduced before. The beta-G and Kw-G families are obtained from (2.23) by choosing the identity function for $W[G(x)]$ and the beta and Kumaraswamy PDFs (with support in $(0, 1)$) for $r(t)$, respectively. The Γ -G family is just defined by $G[W(x)] = -\log[1 - G(x)]$ and taking the gamma PDF with unity scale parameter and shape parameter $a > 0$ for $r(t)$.

We can generate thousands of distributions from Eq.(2.24) by choosing the density $r(t)$ and the function $W[G(x)]$ appropriately. Some of the basic motivations for selecting the density (2.24) in practice are as follows: (i) construct heavy-tailed distributions that are not longer-tailed for modeling real data; (ii) make the kurtosis more flexible compared to the baseline model; (iii) produce a skewness for symmetrical distributions; (iv) generate distributions with symmetric, left-skewed, right-skewed, and reversed J-shaped; (v) provide consistently better fits than other generated models under the same baseline distribution; and (vi) define special models with all types of the HRF.

Let $G(x)$, $g(x)$, $\bar{G}(x) = 1 - G(x)$, and $Q_G(p) = G^{-1}(p)$ be the CDF, PDF, SF, and QF of any rv X . We can define the odds (O), log-odds (LO), and log-odds ratio (LOR) functions by $O(x) = [G(x)/\bar{G}(x)]$, $LO(x) = \log [G(x)/\bar{G}(x)]$, and $LOR(x) = g(x)/[G(x)\bar{G}(x)]$, respectively. The use of the odds ratio is becoming very popular and has applications in the fields of reliability and survival analysis, large sample theory, and discriminant analysis, among others. The LOR is also a useful measure for modeling data that exhibits non-monotone failure rate. The distributions being non-monotone in terms of failure rate are monotone in terms of LOR.

Some generalized classes have been proposed using the O-function in the statistical literature. For each baseline G , the CDF of the odd log-logistic-G (OLL-G) class follows from (2.23) by integrating the log-logistic PDF

$$F(x) = \int_0^{\frac{G(x)}{1-G(x)}} \frac{\alpha t^{\alpha-1}}{(1+t^\alpha)^2} dt = \frac{G(x)^\alpha}{G(x)^\alpha + [1-G(x)]^\alpha}, \quad (2.25)$$

where $\alpha > 0$ is an extra shape parameter to the G distribution. The PDF corresponding to (2.25) is given by

$$f(x) = \frac{\alpha g(x) G(x)^{\alpha-1} [1-G(x)]^{\alpha-1}}{\{G(x)^\alpha + [1-G(x)]^\alpha\}^2}, \quad (2.26)$$

where $g(x)$ is the baseline PDF. This PDF allows greater flexibility of its tails and can be useful in applications in several areas. It will be most tractable when $G(x)$ and $g(x)$ have closed-forms.

We now present a wide distribution in the OLL-G family. The three-parameter generalized gamma distribution (GG) pioneered by Stacy [21] includes as special models the exponential, Weibull, gamma, and Rayleigh distributions, among others. The GG distribution has been used in several research areas such as engineering, hydrology, and survival analysis. It is suitable for modeling data with different forms of HRF: increasing, decreasing, in the form of bathtub, and unimodal. The CDF and PDF of the $GG(\rho, \tau, p)$ distribution is

$$G(x; \rho, \tau, p) = \gamma_1 \left(p, \left(\frac{x}{\rho} \right)^\tau \right) = \frac{\gamma(p, (x/\rho)^\tau)}{\Gamma(p)}, \quad x > 0,$$

$$g(x; \rho, \tau, p) = \frac{\tau}{\rho \Gamma(p)} \left(\frac{x}{\rho} \right)^{\tau p - 1} \exp \left[- \left(\frac{x}{\rho} \right)^\tau \right],$$

where $\rho > 0$, $\tau > 0$, $p > 0$, and $\gamma_1(p, x) = \gamma(p, x)/\Gamma(p)$ is the incomplete gamma function ratio.

The OLLGG density (for $x > 0$) with four positive parameters is defined by substituting $G(x; \rho, \tau, p)$ and $g(x; \rho, \tau, p)$ into Eq. (2.26)

$$f(x) = \frac{\alpha \tau (x/\rho)^{\tau p - 1} \exp[-(x/\rho)^\tau] \{\gamma_1(p, (x/\rho)^\tau) [1 - \gamma_1(p, (x/\rho)^\tau)]\}^{\alpha-1}}{\rho \Gamma(p) \{\gamma_1(p, (x/\rho)^\tau)^\alpha + [1 - \gamma_1(p, (x/\rho)^\tau)]^\alpha\}^2},$$

where ρ is a scale parameter and τ , p , and α are shape parameters.

Some sub-models can be immediately defined from this equation: OLL-gamma ($\tau = 1$), OLL-Weibull ($p = 1$), OLL-exponential ($\tau = p = 1$), OLL-Rayleigh ($\tau = 2$ and $p = 1$), OLL-Maxwell ($\tau = 2$ and $p = 3/2$), and OLL-folded normal ($\tau = \sqrt{2}$ and $p = 1/2$).

One major benefit of the OLLGG density is its ability of fitting skewed data that cannot be properly fitted by existing distributions. It also allows for greater flexibility of its tails and can be widely applied in many areas of engineering and biology.

Finally, we provide an example of (2.23). Let T be a logistic rv with CDF $R(t) = (1 + e^{-\alpha t})^{-1}$ and PDF $r(t) = \alpha e^{-\alpha t} (1 + e^{-\alpha t})^{-2}$ with support in \mathbb{R} , where $\alpha > 0$. By setting $W[G(x)] = \log\{-\log[G(x)]\}$, a monotonically nonincreasing function in $G(x)$, the CDF of the logistic-G family is defined from (2.23) as

$$F(x) = 1 - [1 + \{-\log[G(x)]\}^{-\alpha}]^{-1}, \quad x \in \mathbb{R}.$$

The corresponding PDF takes the form

$$f(x) = \frac{\alpha g(x) \{-\log[G(x)]\}^{-\alpha-1}}{G(x) [1 + \{-\log[G(x)]\}^{-\alpha}]^2}, \quad x \in \mathbb{R}.$$

2.8 Fitting Distributions

All distributions in the families mentioned previously can be fitted to real data sets using the *AdequacyModel* package for the R statistical computing environment (<https://www.r-project.org/>). An important advantage of this package is that it is not necessary to define the log-likelihood function and that it computes the maximum likelihood (ML) estimates, their standard errors, and the formal statistics presented in the next section. We only need to provide the PDF and CDF of the distribution to be fitted to a data set. This *AdequacyModel* package uses the PSO (particle swarm optimization) method obtained by traditional global search approaches such as the quasi-Newton BFGS, Nelder-Mead, and simulated annealing methods to maximize the log-likelihood function. This method does not require initial values. More details are available at <https://rdrr.io/cran/AdequacyModel/>.

2.9 Real Data Illustrations

In this section, we shall illustrate the use of some generalized distributions in practice, specifically the beta-BS and Γ -W distributions. First, we consider an uncensored data set from Nichols and Padgett [18] on breaking stress of carbon fibers (in Gba). Table 2.1 lists the ML estimates of the parameters (standard errors between parentheses), and the values of AIC (Akaike Information Criterion), BIC (Bayesian Information Criterion), and CAIC (Consistent Akaike Information Criterion) are given in Table 2.2. From the values of these statistics, we conclude that the beta-BS model provides a better fit to these data than the BS model. The LR statistic to test the hypothesis H_0 : BS against H_1 : beta-BS is 17.6679 (p -value < 0.01). Thus, using any usual significance level, we reject the null hypothesis in favor of the beta-BS distribution, i.e., the beta-BS distribution is significantly better than the BS distribution. We can also perform formal goodness-of-fit tests in order to verify which distribution fits better to these data. We apply the Cramér-von Mises (CM) and Anderson-Darling (AD) tests. In general, the smaller the values of CM and AD, the better the fit to the data. For the first data set, we have $CM = 0.4601$ and $AD = 2.5896$ for the BS distribution and $CM = 0.2108$ and $AD = 1.2010$ for the beta-BS distribution. Thus, according to these statistics, the second distribution fits these data better than the first distribution.

As a second application, we analyze a real data set on the strengths of 1.5 cm glass fibers, measured at the National Physical Laboratory, England. Unfortunately, the units of measurement are not given in the paper. Table 2.3 lists the ML estimates of the parameters (standard errors between parentheses), and the values of AIC, BIC, and CAIC are given in Table 2.4. Again, from the values of these statistics,

Table 2.1 ML estimates (standard errors in parentheses); first data set

Distribution	Estimates			
	α	β	a	b
Beta-BS	1.0445 (0.0036)	57.6001 (0.3313)	0.1930 (0.0259)	1876.7324 (605.05)
BS	0.4371 (0.0381)	2.5154 (0.1321)		

Table 2.2 AIC, BIC, and CAIC; first data set

Distribution	Statistic		
	AIC	BIC	CAIC
Beta-BS	190.71	199.47	191.36
BS	204.38	208.75	204.57

Table 2.3 ML estimates (standard errors in parentheses); second data set

Distribution	Estimates			
	α	β	a	b
Beta-BS	1.0505 (0.0101)	30.4783 (0.5085)	0.3638 (0.0709)	7857.5658 (3602.2)
BS	0.2699 (0.0267)	1.3909 (0.0521)		

Table 2.4 AIC, BIC, and CAIC; second data set

Distribution	Statistic		
	AIC	BIC	CAIC
Beta-BS	37.552	45.280	38.422
BS	48.378	52.242	48.628

we conclude that the beta-BS model provides a better fit to these data than the BS model. The LR statistic to test the hypothesis H_0 : BS against H_1 : beta-BS is 14.8258 (p -value < 0.01). Thus, the beta-BS distribution is significantly better than the BS distribution. Additionally, we have $CM = 0.6395$ and $AD = 3.3894$ for the BS distribution and $CM = 0.3651$ and $AD = 1.9728$ for the beta-BS distribution. Again, the wider distribution fits the data set better than the baseline distribution.

Finally, we consider the data set consisting of the number of successive failures for the air-conditioning system of each member in a fleet of 13 Boeing 720 jet airplanes; see [19]. The ML estimates of the parameters (standard errors between parentheses) of the Weibull and Γ -W distributions and the values of the AIC, BIC, and CAIC statistics are reported in Tables 2.5 and 2.6, respectively. The values in Table 2.6 indicate that the Γ -W model is better than the usual Weibull model in terms of fitting to these data. The LR statistic to test the hypothesis H_0 : Weibull against H_1 : Γ -W is 7.42 (p -value < 0.01). Thus, we reject the null hypothesis in favor of the Γ -W distribution using any usual significance level. Therefore, the Γ -Weibull distribution is significantly better than the Weibull distribution based on the LR test. Further, we have $CM = 0.0348$ and $AD = 0.2628$ for the Weibull

Table 2.5 ML estimates (standard errors in parentheses); third data set

Distribution	Estimates		
	λ	c	a
Γ -W	1.0915 (1.7942)	0.3735 (0.1739)	5.1353 (4.5227)
Weibull	0.0170 (0.0044)	0.9109 (0.0503)	

Table 2.6 AIC, BIC, and CAIC; third data set

Distribution	Statistic		
	AIC	BIC	CAIC
Γ -W	2072.1	2081.8	2072.2
Weibull	2077.5	2084.0	2077.6

distribution and $CM = 0.1594$ and $AD = 0.9968$ for the Γ -W distribution. These results also indicate that the Γ -W distribution provides a more adequate fit to these data than the Weibull distribution.

References

- Alzaatreh, A., Famoye, F., Lee, C.: A new method for generating families of continuous distributions. *Metron* **71**(1), 63–79 (2013)
- Barreto-Souza, W., Lemonte, A.J., Cordeiro, G.M.: General results for the Marshall and Olkin's family of distributions. *An. Acad. Bras. Cienc.* **85**(1), 3–21 (2013)
- Birnbaum, Z.W., Saunders, S.C.: A new family of life distributions. *J. Appl. Probab.* **6**(2), 319–327 (1969)
- Castallares, F., Lemonte, A.J.: A new generalized Weibull distribution generated by gamma random variables. *J. Egypt. Math. Soc.* **23**(2), 382–390 (2015)
- Cordeiro, G.M., de Castro, M.: A new family of generalized distributions. *J. Stat. Comput. Simul.* **81**(7), 883–898 (2011)
- Eugene, N., Lee, C., Famoye, F.: Beta-normal distribution and its applications. *Commun. Stat. Theory and Methods* **31**(4), 497–512 (2002)
- Gupta, R.D., Kundu, D.: Exponentiated exponential family: an alternative to gamma and Weibull distributions. *Biom. J.* **43**(1), 117–130 (2001)
- Gupta, R.C., Gupta, R.D., Gupta, P.L.: Modeling failure time data by Lehmann alternatives. *Commun. Stat. Theory and Methods* **27**(4), 887–904 (1998)
- Lemonte, A.J.: The beta log-logistic distribution. *Brazilian Journal of Probability and Statistics* **28**(3), 313–332 (2014)
- Lemonte, A.J., Cordeiro, G.M., Moreno-Arenas, G.: A new useful three-parameter extension of the exponential distribution. *Statistics* **50**(2), 312–337 (2016)
- Marshall, A.W., Olkin, I.: A new method for adding a parameter to a family of distributions with application to the exponential and Weibull families. *Biometrika* **84**(3), 641–652 (1997)
- Mudholkar, G.S., Srivastava, D.K.: The exponentiated Weibull family: a reanalysis of the bus-motor-failure data. *Technometrics* **37**(4), 436–445 (1995)
- Nadarajah, S., Gupta, A.K.: The exponentiated gamma distribution with application to drought data. *Calcutta Statist. Assoc. Bull.* **59**(1–2) 29–54 (2007)
- Nadarajah, S., Haghighi, F.: An extension of the exponential distribution. *Statistics* **45**(6), 543–558 (2011)
- Nadarajah, S., Kotz, S.: The exponentiated-type distributions. *Acta Appl. Math.* **92**(2), 97–111 (2006)
- Nadarajah, S., Cordeiro, G.M., Ortega, E.M.M.: Some general results for the beta-modified Weibull distribution. *J. Stat. Comput. Simul.* **81**(10), 1211–1232 (2011)
- Nadarajah, S., Cordeiro, G.M., Ortega, E.M.M.: General results for the Kumaraswamy G distribution. *J. Stat. Comput. Simul.* **87**(7), 951–979 (2012)
- Nichols, M.D., Padgett, W.J.: A Bootstrap control chart for Weibull percentiles. *Qual. Reliab. Eng. Int.* **22**(2), 141–151 (2006)
- Proschan, F.: Theoretical explanation of observed decreasing failure rate. *Technometrics* **5**(3), 375–383 (1963)
- Ristić, M.M., Balakrishnan, N.: The gamma-exponentiated exponential distribution. *J. Stat. Comput. Simul.* **82**(8), 1191–1206 (2012)
- Stacy, E.W.: A Generalization of the Gamma Distribution. *Ann. Math. Stat.* **33**(3), 1187–1192 (1962)
- Tahir, M.H., Nadarajah, S.: Parameter induction in continuous univariate distributions: Well-established G families. *An. Acad. Bras. Cienc.* **87**(2), 539–568 (2015)
- Zografos, K., Balakrishnan, N.: On families of beta-and generalized gamma-generated distribution and associate inference. *Stat. Methodol.* **6**(4), 344–362 (2009)



Gauss M. Cordeiro received his PhD in Statistics at Imperial College of Science and Technology, University of London, in 1982. Currently, Cordeiro is a Class A researcher of the Brazilian Research Council-CNPq, full professor at Federal University of Pernambuco (Brazil) and Member of the Graduate Program in Statistics at the same university. Cordeiro's main research interests in statistics include asymptotic theory, probability distributions and regression models. He has published more than 430 papers in international statistical journals.



Artur Lemonte received his Ph.D. from the University of São Paulo at São Paulo in 2010. He is a professor at Federal University of Rio Grande do Norte, Department of Statistics, Natal/RN, Brazil. He works on higher-order asymptotics, mathematical statistics, regression models, parametric inference, and distribution theory. Up to date, he has published more than 100 papers in refereed statistical journals.



Paul Kvam and Jye-Chyi Lu

Contents

3.1	Introduction and Literature Review	53
3.2	Lifetime Distributions in Reliability	54
3.2.1	Alternative Properties to Describe Reliability.....	55
3.2.2	Conventional Reliability Lifetime Distributions.....	55
3.2.3	From Physics to Failure Distributions.....	55
3.2.4	Lifetime Distributions from Degradation Modeling.....	56
3.2.5	Censoring.....	57
3.2.6	Probability Plotting.....	57
3.3	Analysis of Reliability Data	57
3.3.1	Maximum Likelihood.....	58
3.3.2	Likelihood Ratio.....	58
3.3.3	Kaplan–Meier Estimator.....	59
3.3.4	Degradation Data.....	60
3.4	System Reliability	61
3.4.1	Estimating System and Component Reliability.....	62
3.4.2	Stochastic Dependence Between System Components.....	62
3.4.3	Logistics Systems.....	63
3.4.4	Robust Reliability Design in the Supply Chain.....	64
	References	64

Abstract

This chapter provides a short summary of fundamental ideas in reliability theory and inference. The first part of the chapter accounts for lifetime distributions that are used in engineering reliability analysis, including general properties of reliability distributions that pertain to lifetime for manufactured products. Certain distributions are formulated on the basis of simple physical properties, and other are more or less empirical. The first part of the

chapter ends with a description of graphical and analytical methods to find appropriate lifetime distributions for a set of failure data.

The second part of the chapter describes statistical methods for analyzing reliability data, including maximum likelihood estimation (both parametric and nonparametric) and likelihood ratio testing. Degradation data are more prevalent in experiments in which failure is rare and test time is limited. Special regression techniques for degradation data can be used to draw inference on the underlying lifetime distribution, even if failures are rarely observed.

The last part of the chapter discusses reliability for systems. Along with the components that comprise the system, reliability analysis must take account of the system configuration and (stochastic) component dependencies. System reliability is illustrated with an analysis of logistics systems (e.g., moving goods in a system of product sources and retail outlets). Robust reliability design can be used to construct a supply chain that runs with maximum efficiency or minimum cost.

Keywords

Weibull distribution · System reliability · Empirical likelihood · Residual life · Lifetime data

3.1 Introduction and Literature Review

In everyday use, words like reliability and quality have meanings that vary depending on the context. In engineering, *reliability* is defined as the ability of an item to perform its function, usually measured in terms of probability as a function of time. *Quality* denotes how the item conforms to its specifications, so reliability is a measure of the item's quality over time.

P. Kvam (✉)
Department of Mathematics and Computer Science, University of Richmond, Richmond, VA, USA
e-mail: pkvam@richmond.edu

J.-C. Lu
The School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA
e-mail: JCLU@isye.gatech.edu

Since the time of *Birnbaum* and *Sanders* [1], when system reliability emerged as its own discipline, research has centered on the operation of simple systems with identical parts working independently of each other. Today's systems do not fit this mold; system representation must include multifaceted components with several component states that can vacillate between perfect operation and terminal failure. Not only do components interact within systems, but many systems are dynamic in that the system configuration can be expected to change during its operation, perhaps due to component failures or external stresses. Computer software, for example, changes its failure structure during the course of design, testing, and implementation.

Statistical methods for reliability analysis grew from this concept of system examination, and system reliability is often gauged through component lifetime testing. This chapter reviews the current framework for statistical reliability and considers some modern needs from experimenters in engineering and the physical sciences.

Statistical analysis of reliability data in engineering applications cannot be summarized comprehensively in a single book chapter such as this. The following books (listed fully in the reference section) serve as an excellent basis for a serious treatment of the subject:

1. *Statistical Theory of Reliability and Life Testing* by Barlow and Proschan [2]
2. *Practical Methods for Reliability Data Analysis* by Ansell and Phillips [3]
3. *Reliability: Probabilistic Models and Statistical Methods* by Leemis [4]
4. *Applied Reliability* by Tobias and Trindade [5]
5. *Engineering Reliability* by Barlow [6]
6. *Reliability for Technology, Engineering, and Management* by Kales [7]
7. *Statistical Methods for Reliability Data* by Meeker and Escobar [8]
8. *Reliability Modeling, Prediction, and Optimization* by Blischke and Murthy [9]
9. *Statistical Methods for the Reliability of Repairable Systems* by Rigdon and Basu [10] and
10. *Bayesian Reliability* by Hamada et al. [11]

Some of the books in this list focus on reliability theory, and others focus exclusively on reliability engineering. From the more inclusive books, [8] provides a complete, high-level guide to reliability inference tools for an engineer, and most examples have an engineering basis (usually in manufacturing). For reliability problems closely associated with materials testing, *Bogdanoff* and *Kozin* [12] connect the physics of degradation to reliability models. *Sobczyk* and *Spencer* [13] also relate fatigue to reliability through probability modeling. For reliability prediction in software

performance, *Lyu* [14] provides a comprehensive guide of engineering procedures for software reliability testing, while a more theoretical alternative by *Singpurwalla* and *Wilson* [15] emphasizes probability modeling for software reliability, including hierarchical Bayesian methods. Closely related to reliability modeling in engineering systems, *Bedford* and *Cooke* [16] covers methods of probabilistic risk assessment, which is an integral part of reliability modeling for large and complex systems.

Other texts emphasize reliability assessment in a particular engineering field of interest. For statistical reliability in geotechnical engineering, *Baecher* and *Christian* [17] is recommended as it details statistical problems with soil variability, autocorrelation (i.e., Kriging), and load/resistance factors. *Ohring* [18] provides a comprehensive guide to reliability assessment for electrical engineering and electronics manufacturing, including reliability pertaining to degradation of contacts (e.g., crack growth in solder), optical fiber reliability, semiconductor degradation, and mass-transport-induced failure. For civil engineering, *Melchers'* [19] reliability text has a focus on reliability of structural systems and loads, time-dependent reliability, and resistance modeling.

3.2 Lifetime Distributions in Reliability

While engineering studies have contributed a great deal of the current methods for reliability life testing, an equally great amount exists in the biological sciences, especially relating to epidemiology and biostatistics. Life testing is a crucial component to both fields, but the bio-related sciences tend to focus on mean lifetimes and numerous risk factors. Engineering methods, on the other hand, are more likely to focus on upper (or lower) percentiles of the lifetime distribution as well as the stochastic dependencies between working components. Another crucial difference between the two application areas is that engineering models are more likely to be based on principles of physics that lead to well-known distributions such as Weibull, log-normal, extreme value, and so on.

The failure time distribution is the most widely used probability tool for modeling product reliability in science and industry. If $f(x)$ represents the probability density function for the product's failure time, then its reliability is $R(x) = \int_x^\infty f(u)du$, and $R(t) = 1 - F(t)$ where F is the cumulative distribution function (CDF) corresponding to f . A *quantile* is the CDF's inverse; the p th quantile of F is the lifetime value t_p such that $F(t_p) = p$. To understand the quality of a manufactured product through these lifetime probability functions, it is often useful to consider the notion of *aging*. For example, the (conditional) reliability of a product that has been working t units of time is

$$R(x|t) = \frac{R(t+x)}{R(t)}, \text{ if } R(t) > 0. \quad (3.1)$$

The rate of change of $R(x|t)$ is an important metric for judging a product's quality, and the conditional *failure rate* function $h(t)$ is defined as

$$h(t) = \lim_{x \rightarrow \infty} x^{-1} \frac{R(t) - R(t+x)}{R(t)} = \frac{f(t)}{R(t)}. \quad (3.2)$$

The *cumulative failure rate* (sometimes called the *hazard function*) is $H(t) = \int_0^t h(u)du$, and has many practical uses in reliability theory because of its monotonicity and the fact that $H(t) = -\log R(t)$.

The failure rate clearly communicates how the product ages during different spans of its lifetime. Many manufactured products have an increasing failure rate, but the rate of increase is rarely stable throughout the product's lifetime. If $r(t)$ remains constant, it is easy to show the lifetime distribution is exponential $f(x) = \theta \exp(-\theta x)$, $x > 0$ and the product exhibits no aging characteristics. Many electronic components and other manufactured items have brief initial period when failure rate is relatively high and decrease toward a steady state, where it stays until aging causes the rate to increase. This is called a bath-tub failure rate. The period in which early failures occur (called *infant mortality*) is called the *burn-in* period, and is often used by manufacturers to age products and filter out defectives (early failures) before being making it available to the consumer.

3.2.1 Alternative Properties to Describe Reliability

The failure rate function, reliability function, cumulative hazard function, and probability density describe different aspects of a lifetime distribution. The expected lifetime, or *mean time to failure* (MTTF), is an important measure for repairable systems. Several alternatives for characterizing properties of the lifetime distribution include:

- *Mean residual life* = $L(t) = E_X(X - t|X \geq t)$ which is the expected residual life of a component that has already lasted t units of time. If $L(t)$ is less than the expected lifetime μ , the product is exhibiting aging by the time t .
- *Reversed hazard rate* = $v(t) = f(x)/F(x)$ that provides a different aspect of reliability: the conditional failure frequency at the time just before t given that the product failed in $(0, t]$ (see Chap. 1 of [20], for example).
- *Percentile residual life* = $Q_\alpha = F^{-1}[1 - (1 - \alpha) \times R(t)] - t$ which is the α quantile of the residual life (the conditional lifetime distribution given that the product has lasted t units of time). The median residual life, where $\alpha = 1/2$ compares closely to $L(t)$.

- *Mill's ratio* = $R(x)/f(x) = 1/h(x)$, used in economics, which is not an ordinary way to characterize reliability, but it is worth noting because of its close connection to failure rate.

3.2.2 Conventional Reliability Lifetime Distributions

So far, only one distribution (exponential) has been mentioned. Rather than presenting a formal review of commonly used reliability distributions, a summary of commonly applied lifetime distributions is presented in Table 3.1, including the exponential, gamma, Weibull, log-normal, logistic, Pareto, and extreme value. In the table, $\Gamma(t) = \int_0^\infty x^{t-1} e^{-x} dx$ is the ordinary gamma function, and $IG(t, x)$ represents the corresponding incomplete gamma function.

For manufacturing centers and research laboratories that conduct lifetime tests on products, lifetime data is an essential element of reliability analysis. However, a great deal of reliability analysis is based on field data, or reliability information sampled from day-to-day usage of the product. In many of these instances, lifetime data is a luxury not afforded to the reliability inference. Instead, historical event data and inspection counts are logged for the data analysis. Consequently, several discrete distributions (e.g., Poisson, binomial, and geometric) are important in reliability applications. Chapter ▶ 4 has a more detailed discussion of these and other statistical distributions applied in engineering problems.

3.2.3 From Physics to Failure Distributions

Many of the distributions in Table 3.1 are derived based on physical principles. For example, *Weibull* [21] derived the distribution that takes his name to represent the breaking strength of materials based on the idea that some components are comparable to a chain that is no stronger than its weakest link. From this premise, the distribution can be derived from properties of minimums, in contrast to the extreme value distribution, which can be derived through the properties of maximums (see [22], for example). In a short time after its introduction, the Weibull distribution was successfully applied to numerous modeling problems in engineering and has become the hallmark distribution in applied reliability. A primary reason for its suitability to lifetime analysis is its flexible failure rate; unlike other distributions listed in Table 3.1, the Weibull failure rate is simple to model, easy to demonstrate, and it can be either increasing or decreasing. A mixture of two Weibull distributions can be used to portray a bath-tub failure rate (as long as only one of the shape parameters is less than one). *Mudholkar* et al. [23] introduce

Table 3.1 Common lifetime distributions used in reliability data analysis

Distribution	$f(t), t > 0$	$h(t)$	μ	σ^2	Parameter space
Exponential	$\theta e^{-\theta t}$	θ	$1/\theta$	$1/\theta^2$	$\theta > 0$
Weibull	$\lambda \kappa t^{\kappa-1} e^{-\lambda t^\kappa}$	$\lambda \kappa t^{\kappa-1}$	$\lambda^{-1/\kappa} \Gamma\left(1 + \frac{1}{\kappa}\right)$	$\lambda^{-2/\kappa} \left[\Gamma\left(1 + \frac{2}{\kappa}\right) - \Gamma^2\left(1 + \frac{1}{\kappa}\right) \right]$	$\kappa > 0, \lambda > 0$
Gamma	$\lambda^r \Gamma^{-1}(r) t^{r-1} e^{-\lambda t}$	$\frac{\lambda^r t^{r-1} e^{-\lambda t}}{\Gamma(r)[1 - \Gamma(r, \lambda t)]}$	r/λ	r/λ^2	$r > 0, \lambda > 0$
Log-normal	$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\log t - \mu)^2}{2\sigma^2}}$	$f(t)/R(t)$	$e^{\mu + \sigma^2/2}$	$e^{2\mu + 2\sigma^2} - e^{2\mu + \sigma^2}$	$-\infty < \mu < \infty, \sigma > 0$
Logistic	$\frac{e^{-(t-\lambda)/\beta}}{\beta(1 + e^{-(t-\lambda)/\beta})^2}$	$[\beta(1 + e^{-(t-\lambda)/\beta})]^{-1}$	λ	$(\beta\pi)^2/3$	$-\infty < \lambda < \infty, \beta > 0$
Pareto	$\frac{m\theta^m}{t^{m+1}}$	$\frac{m}{t}$	$\frac{m\theta}{m-1}$	$\frac{m\theta^2}{(m-1)^2(m-2)}$	$t > \theta, m > 0$
Extreme value	$\frac{\exp[-(t-a)/b]}{b \exp[-\exp(-(t-a)/b)]}$	$\frac{\exp[-(t-a)/b]}{b \exp[-\exp(-(t-a)/b)] - 1}$	$a - b\Gamma'(1)$	$(b\pi)^2/6$	$-\infty < a < \infty, b > 0$

a new shape parameter to a generalized Weibull distribution that allows bath-tub-shaped failure rates as well as a broader class of monotone failure rates.

For materials exposed to constant stress cycles with a given stress range, lifetime is measured in number of cycles until failure (N). The Whöler curve (or S - N curve) relates stress level (S) to N as $NS^b = k$, where b and k are material parameters (see [13] for examples). By taking logarithms of the S - N equation, we can express cycles to failure as a linear function: $Y = \log N = \log k - b \log S$. If N is log-normally distributed, then Y is normally distributed and regular regression models can be applied for predicting cycles to failure (at a given stress level). In many settings, the log-normal distribution is applied as the failure time distribution when the corresponding degradation process is based on rates that combine multiplicatively. Despite having a concave-shaped (or upside-down bath-tub shape) failure rate, the log-normal is especially useful in modeling fatigue crack growth in metals and composites.

Birnbaum and Saunders [1] modeled the damage to a test item after n cycles as $B_n = \zeta_1 + \dots + \zeta_n$, where ζ_i represents the damage amassed in the i th cycle. If failure is determined by B_n exceeding a fixed damage threshold value B^* , and if the ζ_i are identically and independently distributed,

$$P(N \leq n) = P(B_n > B^*) \approx \Phi\left(\frac{B^* - n\mu}{\sigma\sqrt{n}}\right), \quad (3.3)$$

where Φ is the standard normal CDF. This happens because B_n will be approximately normal if n is large enough. The reliability function for the test unit is

$$R(t) \approx \Phi\left(\frac{B^* - n\mu}{\sigma\sqrt{n}}\right) \quad (3.4)$$

which is called the *Birnbaum-Saunders* distribution. It follows that

$$W = \frac{\mu\sqrt{N}}{\sigma} - \frac{B^*}{\sigma\sqrt{N}} \quad (3.5)$$

has a normal distribution, which leads to accessible implementation in lifetime modeling (see [24] or [12] for more properties).

3.2.4 Lifetime Distributions from Degradation Modeling

These examples show how the product's lifetime distribution can be implied by knowledge of how it degrades in time. In general, degradation measurements have great potential to improve lifetime data analysis, but they also introduce new problems to the statistical inference. Lifetime models have been researched and refined for many manufactured products that are put on test. On the other hand, degradation models tend to be empirical (e.g., nonparametric) or based on simple physical properties of the test item and its environment (e.g., the Paris crack law, Arrhenius rule, and power law) which often lead to obscure lifetime models. *Meeker and Escobar* [8] provide a comprehensive guide to degradation modeling, and show that many valid degradation models will not yield lifetime distributions with closed-form solutions. Functions for estimating parameters in reliability models are available in R packages, for example, the "WeibullR" package [25].

In a setting where the lifetime distribution is known, but the degradation distribution is unknown, degradation information does not necessarily complement the available lifetime data. For example, the lifetime data may be distributed as Weibull, but conventional degradation models will contradict the Weibull assumption (actually, the rarely used *reciprocal Weibull* distribution for degradation with a fixed failure threshold leads to Weibull lifetimes).

In selecting a degradation model based on longitudinal measurements of degradation, monotonic models are

typically chosen under the assumption that degradation is a one-way process. In some cases, such as the measured luminosity of light displays (vacuum fluorescent displays and plasma display devices), the degradation is not necessarily monotonic because, during the first phase of product life, impurities inside the light display's vacuum are slowly burned off and luminosity increases. After achieving a peak level, usually before 100 h of use, the light slowly degrades in a generally monotonic fashion. See *Bae and Kvam* [26, 27] for details on the modeling of non-monotonic degradation data. Degradation data analysis is summarized in Sect. 3.3.3.

3.2.5 Censoring

For most products tested in regular use conditions (as opposed to especially harsh conditions), the allotted test time is usually too short to allow the experimenter to witness failure times for the entire set that is on test. When the item is necessarily taken off test after a certain amount of test time, its lifetime is *right censored*. This is also called type I censoring. Type II censoring corresponds to tests that are stopped after a certain number of failures (say k out of n , $1 \leq k \leq n$) occur.

Inspection data are lifetimes only observed at fixed times of inspection. If the inspection reveals a failed test item, it must be *left censored* at that fixed time. Items that are still working at the time of the last inspection are necessarily right censored. This is sometimes called *interval censoring*.

Censoring is a common hindrance in engineering applications. Lifetime data that are eclipsed by censoring cause serious problems in the data analysis, but it must be kept in mind that each observation, censored or not, contributes information and increases precision in the statistical inference, overall.

3.2.6 Probability Plotting

Probability plotting is a practical tool for checking the adequacy of a fitted lifetime distribution to a given set of data. The rationale is to transform the observed data according to a given distribution, so a linear relationship exists if the distribution was specified correctly. In the past, probability plotting paper was employed to construct the transformation, but researchers can find plotting options on many computer packages that feature data analysis (e.g., SAS, S-Plus, Matlab, Minitab, and SPSS) making the special plotting paper nearly obsolete. Despite the applicability of this technique, few engineering texts feature in-depth discussion on probability plotting and statistics texts tend to focus on theory more than implementation. *Rigdon and Basu* [10] provide a thorough discussion of basic probability plotting, and

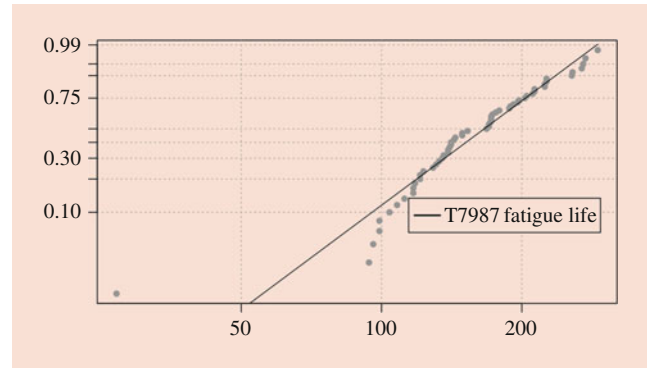


Fig. 3.1 Weibull probability plot for alloy T7987 fatigue life [8]

Atkinson [28] provides a substantial discussion of the subject in the context of regression diagnostics. Advanced plotting techniques even allow for censored observations (see *Waller and Turnbull* [29], for example).

To illustrate how the plot works, we first linearize the CDF of the distribution in question. For example, if we consider the two-parameter Weibull distribution, the *quantile function* is

$$t_p = \left(\frac{-\log p}{\lambda} \right)^{1/\kappa}, \quad (3.6)$$

which implies that the plot of $\log t$ has a linear relationship with the \log - \log function of $p = F(t)$. Hence, Weibull probability plots are graphed on \log - \log probability paper. Figure 3.1 shows a Weibull plot (using Minitab) for the fatigue life of 67 alloy specimens that failed before $n = 300,000$ cycles. This dataset is from *Meeker and Escobar* [8] and the plot also includes 95% confidence bands that identify the uncertainty associated with the plot. In this case, the curvature (especially noticeable on the left side) suggests that the Weibull distribution does not provide an adequate fit.

3.3 Analysis of Reliability Data

Once the lifetime distribution of a test item is determined, the data can be used to estimate important properties of the distribution, including mean, standard deviation, failure rate, reliability (at a fixed time t), and upper or lower quantiles that pertain to early or late failure times.

There are two fundamental methods for approaching the analysis of lifetime data: Bayesian methods and, for the lack of an optimal term, non-Bayesian methods. Although Bayesian methods are accepted widely across many fields of engineering and physical science, non-Bayesian statistics, mostly frequentist and likelihood methods, are still an industry standard. This chapter will not detail how methods of statistical inference are derived in various frameworks of

statistical ideology. Accelerated life testing, an important tool for designing reliability experiments, is discussed in detail in Chap. ▶ 22 and is only mentioned in this chapter. Instead, a summary of important procedures is outlined for statistical estimation, confidence intervals, and hypothesis tests.

3.3.1 Maximum Likelihood

Parametric likelihood methods examine a family of probability distributions and choose the parameter combination that best fits the data. A likelihood function is generally defined by the observed probability model; if the lifetime data X_1, \dots, X_n are independently and identically (IID) distributed with density function $f_X(x; \theta)$, the likelihood function is

$$L(\theta) = \prod_{i=1}^n f_X(x_i; \theta) \quad (3.7)$$

and the *maximum likelihood estimator* (MLE) is the value of θ that maximizes $L(\theta)$. Single-parameter distributions such as the exponential generate easily solved MLEs, but distributions with two or more parameters are not often straightforward. Samples that are not IID lead to complicated likelihood functions and numerical methods are usually employed to solve for MLEs. If an observation x represents a right censoring time, for example, then $P(\text{censor}) = R(x)$ and this information contributes the term $R(x)$ to the likelihood instead of $f(x)$. *Leemis* [4] provides a thorough introduction to the likelihood theory for reliability inference.

For most parametric distributions of interest, the MLE $(\hat{\theta})$ has helpful limit properties. As the sample size $n \rightarrow \infty$, $\sqrt{n}(\hat{\theta} - \theta) \rightarrow N[0, i(\theta)^{-1}]$, where

$$i(\theta) = E \left[\left(\frac{\partial}{\partial \theta} \log f \right)^2 \right] = -E \left(\frac{\partial^2}{\partial \theta^2} \log f \right) \quad (3.8)$$

is the estimator's *Fisher information*. For other parameters of interest, say $\psi(\theta)$, we can construct approximate confidence intervals based on an estimated variance using the Fisher information:

$$\hat{\sigma}^2 [\psi(\hat{\theta})] \approx \frac{1}{\psi(\hat{\theta})^2 i(\hat{\theta})}. \quad (3.9)$$

This allows the analyst to make direct inference for the component reliability $[\psi(\theta; t) = R_\theta(t)$, for example].

Example MLE for failure rate with exponential data (X_1, \dots, X_n) : The likelihood is based on $f(x) = \theta \exp(-\theta x)$ where $\theta > 0$ and is easier to maximize in its natural log form

$$\log L(\theta) = \log \left(\prod_{i=1}^n \theta e^{-\theta x_i} \right) = n \log \theta - \theta \sum_{i=1}^n x_i.$$

The maximum occurs at $\hat{\theta} = 1/\bar{x}$, and the Fisher information $i(\theta) = n/\theta^2$, so an approximate $(1 - \alpha)$ confidence interval is

$$\frac{1}{\bar{x}} \pm z_{\alpha/2} i(\hat{\theta})^{-1/2} = \frac{1}{\bar{x}} \pm z_{\alpha/2} \frac{\hat{\theta}}{\sqrt{n}} = \frac{1}{\bar{x}} \pm z_{\alpha/2} (\bar{x}\sqrt{n})^{-1}. \quad (3.10)$$

In this case, the approximation above is surpassed by an exact interval that can be constructed from the statistic $2\theta(X_1 + \dots + X_n)$ which has a chi-squared distribution with $2n$ degrees of freedom. The confidence statement

$$P[\chi_{2n}^2(1 - \alpha/2) \leq (X_1 + \dots + X_n) \leq \chi_{2n}^2(\alpha/2)] = 1 - \alpha,$$

where $\chi_{2n}^2(\alpha)$ represents the α quantile of the chi-squared distribution with $2n$ degrees of freedom, leads to a $1 - \alpha$ confidence interval for θ of

$$\left(\frac{\chi_{2n}^2(1 - \alpha/2)}{2n\bar{x}}, \frac{\chi_{2n}^2(\alpha/2)}{2n\bar{x}} \right). \quad (3.11)$$

3.3.2 Likelihood Ratio

Uncertainty bounds, especially for multidimensional parameters, are more directly computed using the likelihood ratio (LR) method. Here we consider θ to have p components. Confidence regions are constructed by actual contours (in p -dimensions) of the likelihood function. Define the LR as

$$\Lambda(\theta, \hat{\theta}) = \frac{L(\theta)}{L(\hat{\theta})}, \quad (3.12)$$

where $\hat{\theta}$ is the MLE of L . If θ is the true value of the parameter, then

$$-2 \log \Lambda \sim \chi_p^2,$$

where χ_p^2 is the chi-squared distribution with p degrees of freedom. A $(1 - \alpha)$ confidence region for θ is

$$\left\{ \theta : -2 \log \Lambda(\theta, \hat{\theta}) \leq \chi_p^2(\alpha) \right\}, \quad (3.13)$$

where $\chi_p^2(\alpha)$ represents the $1 - \alpha$ quantile of the χ_p^2 distribution.

Example Confidence region for Weibull parameters: In this case, the MLEs for $\theta = (\lambda, r)$ must be computed using numerical methods. Many statistical software packages

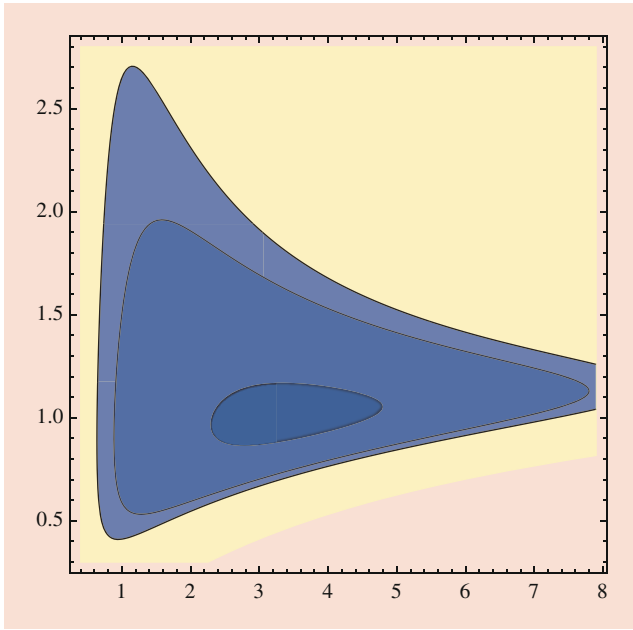


Fig. 3.2 $1 - \alpha = 0.50, 0.90,$ and 0.95 confidence regions for Weibull parameters (λ, r) based on simulated data of size $n = 100$

compute such estimators along with confidence bounds. With $(\hat{\lambda}, \hat{r})$, $L(\hat{\lambda}, \hat{r})$ standardizes the likelihood ratio so that $0 \leq \Lambda(\theta, \hat{\theta}) \leq 1$ and Λ peaks at $(\lambda, r) = (\hat{\lambda}, \hat{r})$. Figure 3.2 shows 50%, 90%, and 95% confidence regions for the Weibull parameters based on a simulated sample of $n = 100$.

Empirical likelihood provides a powerful method for providing confidence bounds on parameters of inference without necessarily making strong assumptions about the lifetime distribution of the product (i.e., it is *nonparametric*). This chapter cannot afford the space needed to provide the reader with an adequate description of its method and theory; *Owen* [30] provides a comprehensive study of empirical likelihood including its application to lifetime data. Rather, we will summarize the fundamentals of nonparametric reliability estimation below.

3.3.3 Kaplan–Meier Estimator

For many modern reliability problems, it is not possible to find a lifetime distribution that adequately fits the available failure data. As a counterpart to parametric maximum likelihood theory, the nonparametric likelihood, as defined by Kiefer and Wolfowitz [31], circumvents the need for continuous densities:

$$L(F) = \prod_{i=1}^n (F(x_i) - F(x_i^-)),$$

where F is the cumulative distribution of the sample X_1, \dots, X_n . Kaplan and Meier [32] developed a nonparametric maximum likelihood estimator for F that allows for censored data in the likelihood. The prevalence of right censoring (when a reliability test is stopped at a time t so the component’s failure time is known only to be greater than t) in reliability studies, along with the increasing computing capabilities provided to reliability analysts, has made nonparametric data analysis more mainstream in reliability studies.

Suppose we have a sample of possibly right-censored lifetimes. The sample is denoted $\{(X_i, \delta_i), i = 1, \dots, n\}$, where X_i is the time measurement, and δ_i indicates whether the time is the observed lifetime ($\delta_i = 1$) or a right censor time $\delta_i = 0$. The likelihood

$$L(F) = \prod_{i=1}^n (1 - F(x_i))^{1-\delta_i} dF(x_i)^{\delta_i}$$

can be shown to be maximized by assigning probability mass only to the observed failure times in the sample according to the rule

$$\hat{F}(t) = 1 - \prod_{x_j \leq t} \left(1 - \frac{d_j}{m_j}\right)$$

where d_j are the number of failures at time x_j and m_j are the number of test items that had survived up to the time x_j^- (i.e., just before the time x_j). A pointwise (approximate) confidence interval can be constructed based on the Kaplan–Meier variance estimate

$$\hat{\sigma}^2(t_i) = \left(1 - \hat{F}(t_i)\right)^2 \sum_{t_i \leq t_j} \frac{d_j}{m_j(m_j - d_j)}.$$

Nair [33] showed that large-sample approximations work well in a variety of settings, but for medium samples, the following serves as an effective $(1 - \alpha)$ -level confidence interval for the survival function $1 - F(t)$:

$$\left(1 - \hat{F}(t)\right) \pm \sqrt{-\frac{\ln(\frac{\alpha}{2})}{2n}} \left(1 - \hat{F}(t)\right)^2 \left(1 + \hat{\sigma}^2(t)\right).$$

We illustrate in Fig. 3.3 the nonparametric estimator with strength measurements (in coded units) for 48 pieces of weathered cord along with 95% pointwise confidence intervals for the cord strength. The data, found in Crowder, et al. [34], include seven measurements that were damaged and yielded strength measurements that are considered right censored.

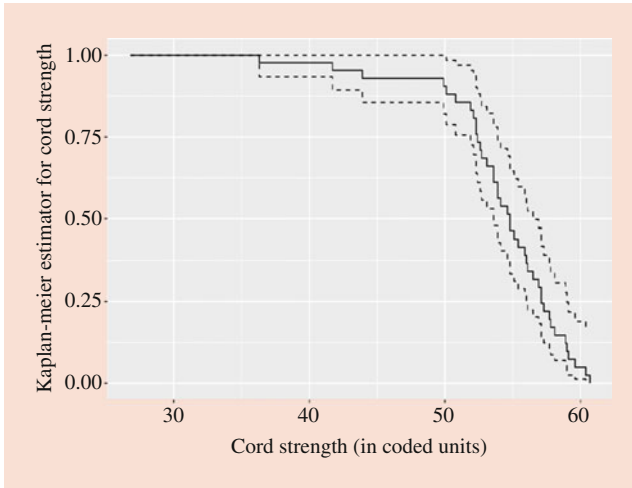


Fig. 3.3 Estimated Kaplan–Meier survival function (1-F) for cord strength data, along with 95% pointwise confidence interval

3.3.4 Degradation Data

As an alternative to traditional life testing, degradation tests can be effective in assessing product reliability when measurements of degradation leading to failure are observable and quantifiable. *Meeker* and *Escobar* [8] provide the most comprehensive discussion on modeling and analyzing degradation data for manufactured items that have either a soft failure threshold (i.e., an arbitrary fixed point at which the device is considered to have failed) or items that degrade before reaching a failed state. In the electronics industry, product lifetimes are far too long to test in a laboratory; some products in the lab will tend to become obsolete long before they actually fail. In such cases, accelerated degradation testing (ADT) is used to hasten product failure. In the manufacture of electronic components, this is often accomplished by increasing voltage or temperature. See Chap. ▶ 22 for a review of recent results in ALT.

If the degradation path is modeled as

$$y_i(t) = \eta_i(t) + \epsilon_i(t), \quad (3.14)$$

where η_i is the path of the i th tested unit ($i = 1, \dots, n$) and ϵ_i represents an error term that has a distribution $H(\epsilon; \Sigma)$ with parameter Σ unknown. Failure would be declared once $y_i(t)$ passes a certain degradation threshold, say y^* . The lifetime distribution can be computed as (assuming degradation is an increasing function)

$$F(t) = P[y(t) > y^*] = P[\epsilon_i(t) > y^* - \eta_i(t)]. \quad (3.15)$$

If η is a deterministic function, the lifetime distribution is driven completely by the error term. This is not altogether realistic. In most cases, item-to-item variability exists and the

function η contains *random coefficients*; that is, $\eta(t) = \eta(t, \lambda, \theta)$, where λ is a vector of unknown parameters (common to all units) and θ is a vector of random coefficients which have a distribution G (with further unknown parameters β) so that realizations of θ change from unit to unit. With an accumulated set of unknown parameters (λ, β, Σ), this makes for a difficult computation of the lifetime distribution. Numerical methods and simulations are typically employed to generate point estimates and confidence statements.

Least squares or maximum likelihood can be used to estimate the unknown parameters in the degradation model. To estimate $F(t_0)$, one can simulate M degradation curves (choosing M to be large) from the estimated regression by generating M random coefficients $\theta_1, \dots, \theta_M$ from the estimated distribution $G(\theta; \hat{\beta})$. Next compute the estimated degradation curve for y_i based on the model with θ_i and $\hat{\lambda}$: $y_i(t) = \eta_i(t; \hat{\lambda}, \theta_i)$. Then $\hat{F}(t_0)$ is the proportion of the M generated curves that have reached the failure threshold y^* by time t_0 .

Meeker and *Escobar* use *bootstrap confidence intervals* for measuring the uncertainty in the lifetime distribution estimate. Their method follows the general algorithm for nonparametric bootstrap confidence intervals described in *Efron* and *Tibshirani* [35]. There are numerous bootstrap sampling methods for various uncertainty problems posed by complex models. This algorithm uses a nonparametric bootstrap sampling procedure which resamples n of the sample degradation curves *with replacement* (i.e., some curves may not be represented in the sample while others may be represented multiple times). This resampled set will be termed the bootstrap sample in the following procedure for constructing confidence intervals.

1. Compute estimates of the parameters β, λ, Σ .
2. Use simulation (as above) to construct $\hat{F}(t_0)$.
3. Generate $N \geq 1000$ bootstrap samples, and for each one, compute estimates $\hat{F}^{(1)}(t_0), \dots, \hat{F}^{(N)}(t_0)$. This is done as before except now the M simulated degradation paths are constructed with an error term generated from $H(\eta; \hat{\Sigma})$ to reflect variability in any single degradation path.
4. With the collection of bootstrap estimates from step 3, compute a $1 - \alpha$ confidence interval for $F(t_0)$ as $[\hat{F}^l(t_0), \hat{F}^u(t_0)]$, where the indexes $1 \leq l \leq u \leq N$ are calculated as $l/N = \Phi[2\Phi^{-1/2}(p_0) + \Phi^{-1/2} \times (\alpha/2)]$ and $u/N = \Phi[2\Phi^{-1/2}(p_0) + \Phi^{-1/2} \times (1 - \alpha/2)]$, and p_0 is the proportion of bootstrap estimates of $F(t_0)$ less than $\hat{F}(t_0)$.

Procedures based on realistic degradation models can obviously grow to be computationally cumbersome, but for important applications the increase in statistical efficiency can be dramatic. In the past, these computations have

impeded degradation analysis from being a feature of reliability problem-solving. Such analyses are easier to implement now, and the reliability analyst need not be coerced into using an overly simplistic model – for instance, a linear model that does not allow for random coefficients.

3.4 System Reliability

A *system* is an arrangement of components that work together for a common goal. So far, the discussion has fixated on the lifetime analysis of a single component, so this represents an extension of single-component reliability study. At the simplest level, a system contains n components of an identical type that are assumed to function independently. The mapping of component outcomes to system outcomes is through the system's structure function. The *reliability function* describes the system reliability as a function of component reliability.

A *series* system is such that the failure of any of the n components in the working group causes the system to fail. If the probability that a single component fails in its mission is p , the probability the system fails is $1 - P(\text{system succeeds}) = 1 - P(\text{all } n \text{ components succeed}) = 1 - (1 - p)^n$. More generally, in terms of component reliabilities (p_1, \dots, p_n) , the system reliability function Ψ is

$$\Psi(p_1, \dots, p_n) = \prod_{i=1}^n (1 - p_i). \quad (3.16)$$

A *parallel* system is just the opposite; it fails only after every one of its n working components fail. The system failure probability is then

$$\Psi(p_1, \dots, p_n) = 1 - \prod_{i=1}^n p_i. \quad (3.17)$$

The parallel system and series system are special cases of a k -out-of- n system, which is a system that works as long as at least k out of its n components work. Assuming $p_i = p$, $i = 1, \dots, n$, the reliability of a k -out-of- n systems is

$$\Psi(p) = \sum_{i=k}^n \binom{n}{i} (1 - p)^i p^{n-i}. \quad (3.18)$$

Of course, most component arrangements are much more complex than a series or parallel system. With just three components, there are five unique ways of arranging the components in a coherent way (that is, so that each component success contributes positively to the system reliability). Figure 3.4 shows the *system structure* of those five arrangements

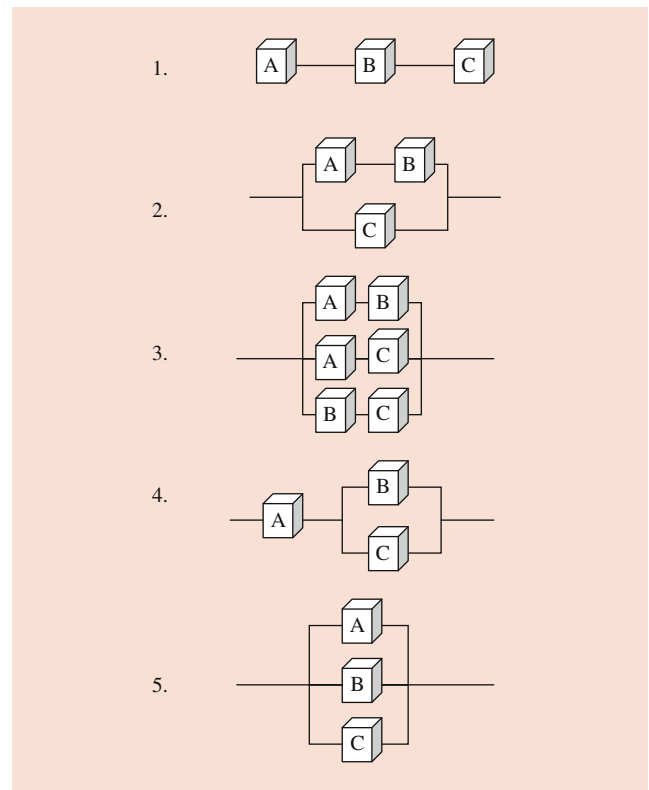


Fig. 3.4 Five unique systems of three components: (1) is series, (3) is 2-out-of-3, and (5) is parallel

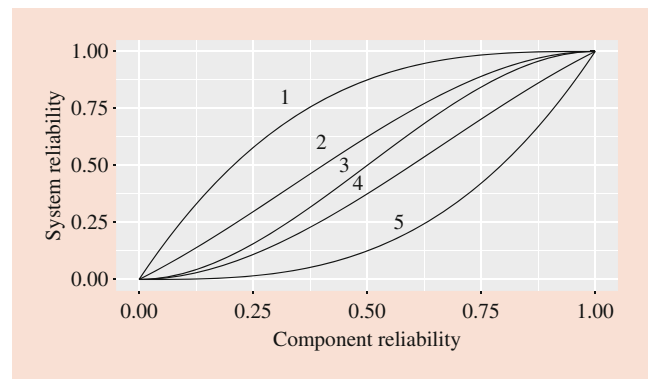


Fig. 3.5 System reliabilities of five system configurations in Fig. 3.4 from the parallel system (1) to the series system (5)

in terms of a *logic diagram* including a series system (1), a 2-out-of-3 system (3), and a parallel system (5). Note that the 2-out-of-3 system cannot be diagrammed with only three components, so each component is represented twice in the logic diagram. Figure 3.5 displays the corresponding reliabilities, as a function of the component reliability $0 \leq p \leq 1$ of those five systems. Fundamental properties of coherent systems are discussed in [2, 4].

3.4.1 Estimating System and Component Reliability

In many complex systems, the reliability of the system can be computed through the reliability of the components along with the system's structure function. If the exact reliability is too difficult to compute explicitly, reliability bounds might be achievable based on *minimum cut sets* (MCS) and *minimum path sets* (MPS). An MPS is the collection of the smallest component sets that are required to work in order to keep the system working. An MCS is the collection of the smallest component sets that are required to fail in order for the system to fail. Table 3.2 shows the minimum cuts sets and path sets for the three-component systems from Fig. 3.4.

In most industrial systems, components have different roles and varying reliabilities, and often the component reliability depends on the working status of other components. System reliability can be simplified through fault-tree analyses (see Chap. 7 of [16], for example), but uncertainty bounds for system reliability are typically determined through simulation.

In laboratory tests, component reliabilities are determined and the system reliability is computed as a function of the statistical inference of component lifetimes. In field studies, the tables are turned. Component manufacturers seeking reliability data outside laboratory tests look to component lifetime data within a working system. For a k -out-of- n system, for example, the system lifetime represents an order statistic of the underlying distribution function. That is, if the ordered lifetimes form a set of independent and identically distributed components ($X_{1:n} \leq X_{2:n} \leq \dots \leq X_{n:n}$), then $X_{n-k+1:n}$ represents the k -out-of- n system lifetime. The density function for $X_{r:n}$ is

$$f_{r:n}(t) = r \binom{n}{r} F(t)^{r-1} [1 - F(t)]^{n-r} f(t), t > 0. \quad (3.19)$$

Kvam and Samaniego [36] derived the nonparametric maximum likelihood estimator for $F(t)$ based on a sample of k -out-of- n system data, and showed that the MLE $\hat{F}(t)$ is consistent. If the i -th system ($i = 1, \dots, m$) observed is a k_i -out-of- n_i system, the likelihood can be represented as

$$L(F) = \prod_{i=1}^m f_{k_i:n_i}(t_i) \quad (3.20)$$

Table 3.2 Minimum cut sets and path sets for the systems in Fig. 3.3

System	Minimum path sets	Minimum cut sets
1	{A,B,C}	{A}, {B}, {C}
2	{A,B}, {C}	{A,C}, {B,C}
3	{A,B}, {A,C}, {B,C}	{A,B}, {A,C}, {B,C}
4	{A,B}, {A,C}	{A}, {B,C}
5	{A}, {B}, {C}	{A,B,C}

and numerical methods are employed to find \hat{F} . Huang [37] investigated the asymptotic properties of this MLE, and Chen [38] provides an ad hoc estimator that examines the effects of censoring.

Compared to individual component tests, observed system lifetimes can be either advantageous or disadvantageous. With an equal number of k -out-of- n systems at each $1 \leq k \leq n$, Takahasi and Wakimoto [39] showed that the estimate of MTTF is superior to that of an equal number of individual component tests. With an unbalanced set of system lifetimes, no such guarantee can be made. If only series systems are observed, Kvam and Samaniego [40] show how the uncertainty in $\hat{F}(t)$ is relatively small in the lower quantiles of F (where system failures are observed) but explodes in the upper quantiles.

3.4.2 Stochastic Dependence Between System Components

Almost all basic reliability theory is based on systems with independently operating components. For realistic modeling of complex systems, this assumption is often impractical; system components typically operate at a level related to the quality and operational state of the other system components.

External events that cause the simultaneous failure of component groups is a serious consideration in reliability analysis of power systems. This can be a crucial point in systems that rely on built-in component redundancy to achieve high target system reliability. Shock models, such as those introduced by Marshall and Olkin [41], can be employed to demonstrate how multiple component failures can occur. An extra failure process is added to the otherwise independent component failure processes, representing the simultaneous failure of one or more components, thus making the component lifetimes positively dependent. This is the basis for most dependent failure models in probabilistic risk assessment, including *common cause failure* models used in the nuclear industry (alpha-factor model, beta-factor model, and binomial failure rate model). See Chapt. 8 of Bedford and Cooke [16] for discussion about how these models are used in risk assessment.

In dynamic systems, where system configurations and component reliability can change after an external event or a failure of one or more of the system components, the shock model approach cannot be applied effectively. In some applications, a load-share model applies. Early applications of the load-share system models were investigated by Daniels [42] for studying the reliability of composite materials in the textile industry. Yarns and cables fail after the last fiber (or wire) in the bundle breaks, thus a bundle of fibers can be considered a parallel system subject to a constant tensile load. An individual fiber fails in time with an individual rate that

depends on how the unbroken fibers within the bundle share the load of this stress. Depending on the physical properties of the fiber composite, this load sharing has different meanings in the failure model. Yarn bundles or untwisted cables tend to spread the stress load uniformly after individual failures which defines an *equal load-share rule*, implying the existence of a constant system load that is distributed equally among the working components.

As expected, a load-sharing structure within a system can increase reliability (if the load distribution saves the system from failing automatically) but reliability inference is hampered even by the simplest model. *Kvam and Pena* [43] show how the efficiency of the load-share system, as a function of component dependence, varies between that of a series system (equivalent to sharing an infinite load) and a parallel system (equivalent to sharing zero load).

3.4.3 Logistics Systems

Numerous studies have examined fundamental problems in network reliability [44], system performance degradation, and workload rerouting for telecommunication, power, and transportation networks [45, 46]. In comparison, the literature on modeling logistics system reliability or performance degradation is scarce. Logistics systems that transport goods, energy (e.g., electricity and gas), water, sewage, money, or information from origins to destinations are critical to every nation's economic prosperity. Unlike the hub in the typical Internet or telecommunication network, where the messages are not mixed together, logistics distribution centers (DCs) tend to mix products from various sources for risk-pooling purposes [47]. Past studies [48] of road network reliability mainly addressed connectivity and travel time reliability. These developments have limited use in providing a first-cut analysis for system-level planning that involves robust logistics network design to meet reliability requirements or supply chain cost and delivery time evaluation for contract decisions [49].

Consider a logistics network consisting of many suppliers providing goods to several DCs, which support store operations to meet customer demands. The reliability of such a network can be evaluated in terms of the probability of delivering goods to stores in a prespecified time limit t_0 . Traveling time in transport routes contains uncertainty, as does the processing time for products shipped through DCs. Random traveling time is a function of routing distances, road and traffic conditions, and possible delays from seaport or security checkpoint inspections. Traveling distance depends on the configuration of logistics networks. Some retail chains use single-layer DCs, but others use multiple-layer

DCs similar to airline hubs (e.g., regional DCs and global DCs) in aggregating various types of goods. Vehicle routing procedures typically involve trucks that carry similar products to several stores in an assigned region. Different products are consolidated in shipment for risk-pooling purposes and to more easily control delivery time and store-docking operations.

When one DC cannot meet the demands from its regional stores (due to demand increase or the DC's limited capability), other DCs provide backup support to maintain the overall network's service reliability. Focusing on the operations between DCs and stores, *Ni et al.* [49] defined the following network reliability as a weighted sum of the individual reliabilities from each DC's operations:

$$r_{\text{system},k}^* = \left[\sum_{i=1, i \neq k}^M d_i P(T_{m,i}^* < t_0) + \sum_{i=1, i \neq k}^M p_i d_k P(T_{m,k,i}^* < t_0) \right] / \sum_{i=1}^M d_i, \quad (3.21)$$

where d_i is the demand aggregated at the i th DC, $T_{m,i}^*$ is the motion time defined as the sum of traveling time from DC $_i$ to its assigned stores (including material processing time at DC $_i$), p_i is the proportion of products rerouted from DC $_k$ through DC $_i$ due to the limited capability in DC $_k$, and $T_{m,j}^*$ is the modified motion time including the rerouted traveling time.

For modeling the aggregated demand d_i and calculating routing distance, *Ni et al.* [49] proposed a multiscale approximation model to quantify demand patterns at spatially located clustered stores. Then, they evaluated product rerouting strategies for maintaining system service reliability, defined in (3.19). Based on the store locations of a major retail chain, several examples show the importance of designing a robust logistics network to limit service reliability degradation when a *contingency* (e.g., multiple DC failure) occurs in the network. Future work includes:

1. Modeling the low-probability but high-impact contingency in the DCs [49] and routes for calculating their relative importance to network reliability.
2. Examining the trade-off between the cost of adding more DCs and the improvement of service reliability.
3. Resolving the *domino effect* when the added workload to DCs after a local DC failure causes further DC failures due to faulty predetermined rules of rerouting to maintain system reliability (e.g., the 2003 electricity blackout in the northeastern region of the USA).

3.4.4 Robust Reliability Design in the Supply Chain

Past studies of robust parameter design [50] focused on product quality issues and assumed that all the controllable variables are under single ownership. Recent outsourcing trends in automobile and electronic manufacturing processes motivate the presentation in this section. In an automobile manufacturing enterprise system, various parts suppliers have control of variables determining quality and reliability. Most of the automobile supply chain systems assemble these parts into a subsystem and then move these systems to other locations owned by different partners for the final system-level assembly and testing. Every segment of the assembly operation controls a subset of variables leading to different levels of system reliability. Because the warranty policy is addressed to all of the part manufacturing and assembly processes in making the final product, it is important to extend the robust parameter design concept to the supply-chain-oriented manufacturing processes.

Supply chain partners have their own operation objectives (e.g., maximize the profit of manufacturing parts to supply several automobile companies). Some of the objectives are aligned to manufacturing a specific type of product, but there are many potential situations with conflicting objectives. When there is no single ownership of all controllable variables in the manufacturing processes, negotiation is needed to resolve potential conflicts. Game theory [51] is commonly used in supply chain contract decisions. Following the framework of *Chen and Lewis* [52], we can decompose the set of controllable variables into a few subsets owned by distinct partners and formulate the objectives of these partners. The supply chain manager can define the product quality and reliability measures and build models to link them to the controllable and uncontrollable variables that are seen in robust parameter design.

Different negotiation situations (e.g., the final product assembly company has more bargaining power than other partners) will lead to distinct levels selected for the controllable variables (see *Charoensiriwath and Lu* [53], for example, in negotiations). As a result, the reliability of the final product can vary. Designing a supply chain system that leads to the most reliable products (with minimum cost) presents an acute challenge, and warranty policies can be designed correspondingly. Because parts and subsystems are made by various partners, warranty responsibilities for certain parts are distributed among partners under the negotiated supply chain contracts.

References

1. Birnbaum, Z.W., Saunders, S.C.: A new family of life distributions. *J. Appl. Probab.* **6**, 319–327 (1969)
2. Barlow, R.E., Proschan, F.: *Statistical Theory of Reliability and Life Testing*. Holt, Rinehart, Austin (1975)
3. Ansell, J.I., Phillips, M.J.: *Practical Methods for Reliability Data Analysis*. Oxford University Press, Oxford (1994)
4. Leemis, L.: *Reliability: Probabilistic Models and Statistical Methods*. Prentice Hall, Englewood (2009)
5. Tobias, P.A., Trindade, D.C.: *Applied Reliability*. CRC, Boca Raton (2011)
6. Barlow, R.E.: *Engineering Reliability*. Society for Industrial and Applied Mathematics, Alexandria (1998)
7. Kales, P.: *Reliability for Technology, Engineering and Management*. Prentice-Hall, Englewood (1998)
8. Meeker, W.Q., Escobar, L.A.: *Statistical Methods for Reliability Data*. Wiley, New York (2014)
9. Blischke, W.R., Murthy, D.N.P.: *Reliability Modeling, Prediction, and Optimization*. Wiley, New York (2000)
10. Rigdon, S.E., Basu, A.P.: *Statistical Methods for the Reliability of Repairable Systems*. Wiley-Interscience, New York (2000)
11. Hamada, M., Wilson, A.G., Reese, C.S., Martz, H.F.: *Bayesian Reliability*. Springer, Berlin/Heidelberg/New York (2008)
12. Bogdanoff, J.L., Kozin, F.: *Probabilistic Models of Cumulative Damage*. Wiley, New York (1985)
13. Sobczyk, K., Spencer, B.F.: *Random Fatigue from Data to Theory*. Academic, Boston (1992)
14. Lyu, M.R.: *Handbook of Software Reliability Engineering*. McGraw Hill, New York (1996)
15. Singpurwalla, N.D., Wilson, S.P.: *Statistical Methods in Software Engineering*. Springer, Heidelberg/Berlin/New York (1999)
16. Bedford, T., Cooke, R.: *Probabilistic Risk Analysis: Foundations and Methods*. Cambridge University Press, London (2001)
17. Baecher, G., Christian, J.: *Reliability and Statistics in Geotechnical Engineering*. Wiley, New York (2003)
18. Ohring, M.: *Reliability & Failure of Electronic Materials & Devices*. Academic, Boston (1998)
19. Melchers, R.E.: *Structural Reliability Analysis and Prediction*. Wiley, New York (1999)
20. Shaked, M., Shanthikumar, J.G.: *Stochastic Orders and their Applications*. Academic, Boston (2007)
21. Weibull, W.: A statistical theory of the strength of material. *Ingenjors Vetenskaps Akademiens Handligar, Stockholm*. **151**, 5–45 (1939)
22. David, H.A., Nagaraja, H.K.: *Order Statistics*. Wiley, New York (2003)
23. Mudholkar, G.S., Srivastava, D.K., Kollia, G.D.: A generalization of the Weibull distribution with application to the analysis of survival data. *J. Am. Stat. Assoc.* **91**, 1575–1583 (1996)
24. Høyland, A., Rausand, M.: *System Reliability Theory Models and Statistical Methods*. Wiley, New York (1994)
25. Silkworth, D., Szymnck, J., Ormerod, J.: Package ‘WeibullR’: Weibull Analysis for Reliability Engineering R Package. <http://www.openreliability.org/weibull-r-weibull-analysis-on-r/> (2018)
26. Bae, S.J., Kvam, P.H.: A nonlinear random coefficients model for degradation testing. *Technometrics*. **46**(4), 460–469 (2004)
27. Bae, S.J., Kvam, P.H.: A change-point analysis for modeling incomplete burn-in for light displays. *IIE Trans.* **38**, 489–498 (2006)

28. Atkinson, A.C.: Plots, Transformations, and Regression: An Introduction to Graphical Methods of Diagnostic Regression Analysis, Oxford Statistical Science Series. Oxford University Press, Oxford (1992)
29. Waller, L.A., Turnbull, B.W.: Probability plotting with censored data. *Am. Stat.* **46**, 5–12 (1992)
30. Owen, A.: Empirical Likelihood. Chapman and Hall, CRC, Boca Raton (2001)
31. Kiefer, J., Wolfowitz, J.: Consistency of the maximum likelihood estimator in the presence of infinitely many incidental parameters. *Ann. Math. Stat.* **27**, 887–906 (1956)
32. Kaplan, E.L., Meier, P.: Nonparametric maximum likelihood estimation from incomplete observations. *J. Am. Stat. Assoc.* **53**, 457–481 (1958)
33. Nair, V.N.: Confidence bands for survival functions with censored data: a comparative study. *Technometrics.* **26**, 265–275 (1984)
34. Crowder, M.J., Kimber, A.C., Smit, R.L., Sweeting, T.J.: Statistical Analysis of Reliability Data. Chapman and Hall, CRC, London (1994)
35. Efron, B., Tibshirani, R.: An Introduction to the Bootstrap. Chapman and Hall, CRC Press, Boca Raton (1993)
36. Kvam, P.H., Samaniego, F.J.: Nonparametric maximum likelihood estimation based on ranked set samples. *J. Am. Stat. Assoc.* **89**, 526–537 (1994)
37. Huang, J.: Asymptotic of the NPML of a distribution function based on ranked set samples. *Ann. Stat.* **25**(3), 1026–1049 (1997)
38. Chen, Z.: Component reliability analysis of k-out-of-n systems with censored data. *J. Stat. Plan. Inference.* **116**, 305–316 (2003)
39. Takahasi, K., Wakimoto, K.: On unbiased estimates of the population mean based on samples stratified by means of ordering. *Ann. Inst. Stat. Math.* **20**, 1–31 (1968)
40. Kvam, P.H., Samaniego, F.J.: On estimating distribution functions using nomination samples. *J. Am. Stat. Assoc.* **88**, 1317–1322 (1993)
41. Marshall, A.W., Olkin, I.: A multivariate exponential distribution. *J. Am. Stat. Assoc.* **62**, 30–44 (1967)
42. Daniels, H.E.: The statistical theory of the strength of bundles of threads. *Proc. R. Soc. Lond. A.* **183**, 405–435 (1945)
43. Kvam, P.H., Peña, E.: Estimating load-sharing properties in a dynamic reliability system. *J. Am. Stat. Assoc.* **100**(469), 262–272 (2004)
44. Ball, M.O.: Computing network reliability. *Oper. Res.* **27**, 823–838 (1979)
45. Sanso, B., Soumis, F.: Communication and transportation network reliability using routing models. *IEEE Trans. Reliab.* **40**, 29–37 (1991)
46. Shi, V.: Evaluating the performability of tactical communications network. *IEEE Trans. Veh. Technol.* **53**, 253–260 (2004)
47. Chen, A., Yang, H., Lo, H., Tang, W.H.: A capacity related reliability for transportation networks. *J. Adv. Transp.* **33**(2), 183–200 (1999)
48. Dandamudi, S., Lu, J.-C.: Competition driving logistics design with continuous approximation methods. Technical Report. <http://www.isye.gatech.edu/apps/research-papers/>
49. Ni, W., Lu, J.-C., Kvam, P.: Reliability modeling in spatially distributed logistics systems. *IEEE Trans. Reliab.* **55**(3), 525–534 (2006)
50. Wu, C.F.J., Hamada, M.: Experiments: Planning, Analysis, and Parameter Design Optimization, p. 503. Wiley, New York (2009)
51. Fudenberg, D., Tirole, J.: Game Theory. MIT Press, Cambridge (2000)
52. Chen, W., Lewis, K.: Robust design approach for achieving flexibility in multidisciplinary design. *AIAA J.* **37**(8), 982–989 (1999)
53. Charoensiriwath, C., Lu, J.-C.: Competition under manufacturing service and retail price. *Econ. Model.* **28**(3), 1256–1264 (2011)



Paul Kvam is a professor of statistics at the University of Richmond. He joined UR in 2014 after 19 years as professor at Georgia Tech and 4 years as scientific staff at the Los Alamos National Laboratory. Dr. Kvam received his B.S. in mathematics from Iowa State University in 1984, an M.S. in statistics from the University of Florida in 1986, and his Ph.D. in statistics from the University of California, Davis in 1991. His research interests focus on statistical reliability with applications to engineering, nonparametric estimation, and analysis of complex and dependent systems. He is a fellow of the American Statistical Association and an elected member of the International Statistics Institute.

Paul Kvam received his Ph.D. from the University of California, Davis, in 1991. He was a staff scientist at Los Alamos National Laboratory from 1991–1995, and professor at Georgia Tech from 1995–2014 in the School of Industrial and Systems Engineering. Kvam is currently professor of statistics in the Department of Mathematics and Computer Science at the University of Richmond.



Jye-Chyi (JC) Lu is a professor in the School of Industrial and Systems Engineering (ISyE). He received a Ph.D. in statistics from University of Wisconsin at Madison in 1988, and joined the faculty of North Carolina State University where he remained until 1999 when he joined Georgia Institute of Technology. Dr. Jye-Chyi Lu's research areas cover industrial statistics, signal processing, semiconductor and electronic manufacturing, data mining, bioinformatics, supply chain management, logistics planning, and nanotechnology. He has about 93 disciplinary and interdisciplinary publications, which have appeared in both engineering and statistics journals. He has served as an associate editor for *Technometrics*, *IEEE Transactions on Reliability*, and *Journal of Quality Technology*.

His work now includes reliability, human-in-the-loop variable selections for large p small n problems, decision, and data analytics.



Yuhang Xu

Contents

4.1	Introduction	68
4.2	Functional Principal Component Analysis	69
4.2.1	Model for FPCA	69
4.2.2	Estimation of Mean Function, Eigenvalues, and Eigenfunctions	70
4.2.3	Estimation of PC Scores	71
4.2.4	Prediction of Trajectories and Confidence Bands	72
4.2.5	Implementation Details	73
4.3	Functional Linear Regression	77
4.3.1	Model for FLR	77
4.3.2	Estimation of Regression Function and R^2	78
4.3.3	Prediction and Confidence Bands	78
4.3.4	Implementation Details	79
4.4	Other Selected Topics in FDA and Software Packages	82
4.4.1	Other Selected Topics in FDA	82
4.4.2	Selected Software Packages	84
References	84

Abstract

This chapter introduces functional data analysis (FDA) and selective topics in FDA, including functional principal component analysis (FPCA) and functional linear regression (FLR), with real data applications using a software package, which is publicly available. The methods in this chapter are based on local polynomial regression, a basic and important smoothing technique in nonparametric and semiparametric statistics. The approaches included in this chapter are not limited to the analysis of dense functional data but can also be used for the analysis of sparse functional/longitudinal data.

In Sect. 4.1, we introduce FDA with some interesting examples of functional data and briefly describe FPCA and FLR. Section 4.2 details FPCA, one of the most important topics and tools in FDA. Topics such as the estimation of mean and covariance functions using non-parametric smoothing, choosing the number of principal components (PC) using subjective and objective methods, and prediction of trajectories are included and illustrated using a publicly available bike-sharing data set. Section 4.3 presents FLR based on FPCA described in Sect. 4.2. FLR is a generalization of traditional linear regression to the case of functional data. It is a powerful tool to model the relationship between functional/scalar response and functional predictors. This section is also illustrated using the same bike-sharing data set. We focus on the case when both response and predictor are functions in this section, but we mentioned other types of FLR topics in Sect. 4.4. Section 4.4 presents a short overview of other selected topics and software packages in FDA. These topics are either about functional data with more complex features than the simple and basic ones included in the previous two sections or about other statistical estimation and inference not covered before. The statistical software packages used in this chapter are written in Matlab and may be appropriate for the analysis of some basic types of functional data but not for others. Section 4.4 described other software packages written in different languages, such as R, and those packages have the flexibility to analyze various problems in functional data and different types of functional data.

Keywords

Confidence band · Functional data · Linear regression · Local polynomial · Longitudinal data · Prediction · Principal components

Y. Xu (✉)
Bowling Green State University, Bowling Green, OH, USA

4.1 Introduction

FDA deals with the analysis of data which are in the form of functions. The functions could be curves, images, or other types of forms, depending on the set on which the functions are recorded. A common type of functional data are curves recorded during a time interval. For example, a well-known example of functional data described in a classic book on FDA [1] is the Canadian weather station data. The data set is about the daily temperature and precipitation data recorded in 35 Canadian weather stations. The daily average temperature data are shown in Fig. 4.1. We notice that, on average, the data look like a parabola; also there are obvious variations among those weather stations, and some patterns of correlation may exist between two time points. In FDA, some of those findings can be described using a univariate mean function and a bivariate correlation function.

FDA has received increasing attention and interests during the past two decades. This is probably due to the development of technology, which enables the collection of functional data in many fields, including medical science, public health, engineering, biological science, environmental science, etc.

For example, in plant breeding and precision agriculture, high-throughput phenotyping technologies have been successfully employed, which enables collecting phenotyping data over time automatically. The characteristics of crops over time are examples of functional data. For example, [2] studied the growth and its dynamics of maize using high-throughput phenotyping data. The sizes of maize plants in the study were observed during a 21-day period of time, and their

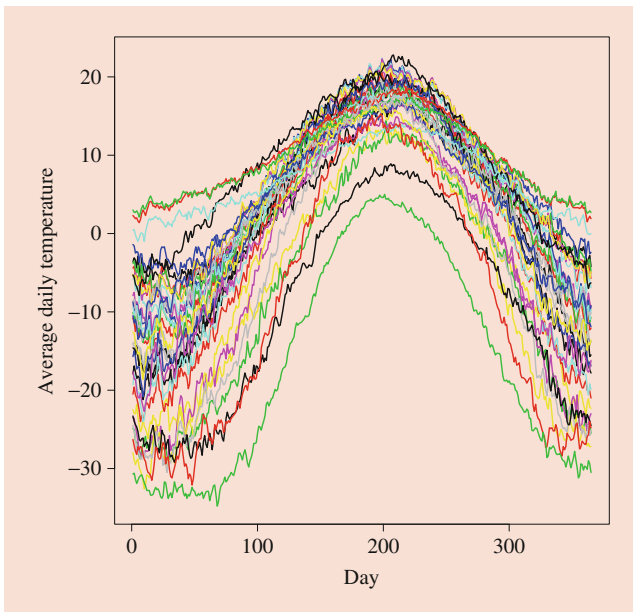


Fig. 4.1 Average daily temperature measured at 35 Canadian weather stations

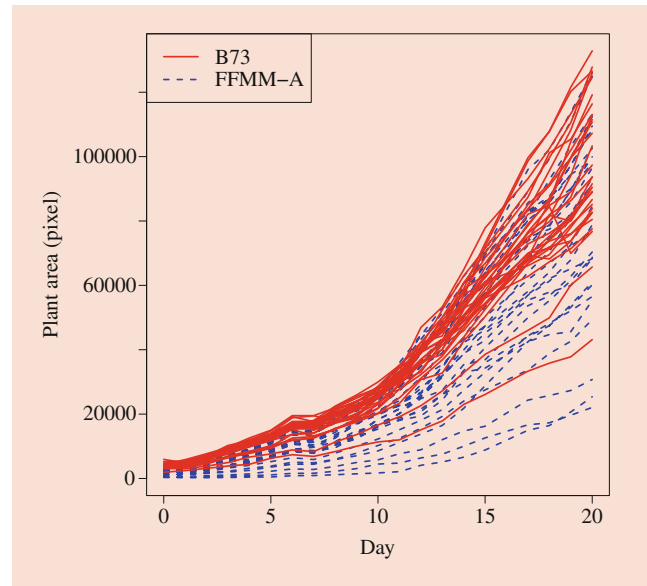


Fig. 4.2 The growth of maize of two genotypes: genotype B73 and genotype FFMM-A

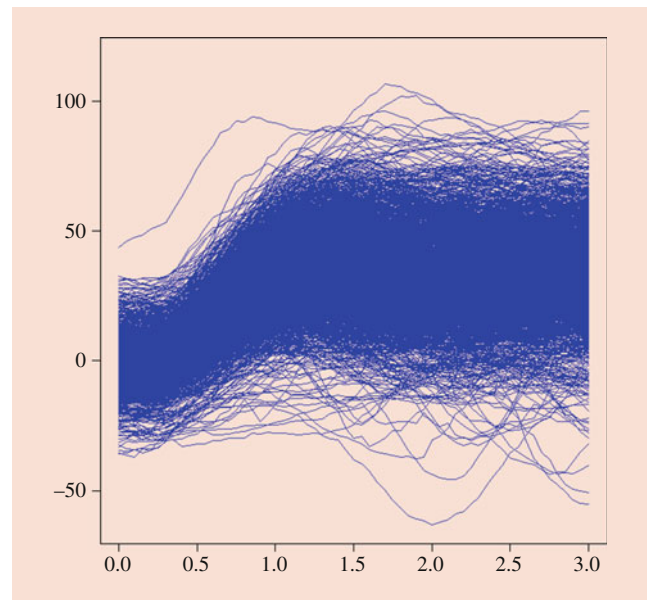


Fig. 4.3 The development of root tip angles of thousands of maize seeds during 3-h experiments

growth curves are shown in Fig. 4.2. In the study of root gravitropism, modern technology also facilitates the collection of gravitropism data automatically using advanced digital cameras. For example, [3] studied the root gravitropism over time using FDA. In particular, the gravitropism data has a multilevel structure, namely, multilevel or hierarchical functional data, which will be discussed in Sect. 4.4.1 (Fig. 4.3).

Functional data are intrinsically infinite-dimensional. They are usually represented by random processes over a fixed time interval. In real-world applications, the processes

are not observed completely but discretely over regular or irregular time points. In classical FDA, e.g., the methods introduced in [1], there are usually abundant observations for each curve, so each curve contains enough information for the trajectory of a subject. For example, in the study of the activities of a brain, data collected using the electro-physiological monitoring method usually have thousands of observations per curve. However, in longitudinal studies, often there are only a few observations per curve, namely, sparse functional data. Yao et al. [4] proposed the idea of pooling information from all curves for the analysis of such type of data. The methods we described in this chapter are mainly based on the idea of [4], but the application of the methods is not limited to sparse functional data.

FPCA, as an important dimension reduction tool, has been used widely in FDA. It captures the majority of the variation in functional data, facilitates the interpretation of the data, and provides outputs which can be used for further statistical analysis. In FPCA, the observed curves are treated as realizations from a random process. The random process is usually decomposed into three parts: an overall fixed mean function, a zero-mean random process, and a zero-mean random variable representing measurement errors. The mean function and the covariance function of the zero-mean process are usually assumed to be smooth. Therefore, non-parametric smoothing methods, such as local polynomial regression and spline-based approaches, become fundamental tools for FPCA. For example, [1] mainly uses spline-based approaches and [4] employs local polynomial regression for smoothing. The main attraction of FPCA is it enables the representation of the zero-mean random process as a finite linear combination of random variables, called functional principal component (PC) scores, and fixed functions, called eigenfunctions.

Probably the most important models in statistics are linear regression models. In a linear regression model, it is assumed that there is a linear relationship between a response and predictors. The counterpart in FDA is called FLR models. In a FLR model, the regression parameter in a classical linear regression model becomes a regression function. The regression function is a one-dimensional function if the response is scalar or a two-dimensional function if the response is also a function. The results of FPCA for the predictor if the response is scalar, or for both the predictor and response if the response is also a function, can be used to obtain the regression function. Similar to classical linear regression models, FLR models can also be used to make a prediction of the unknown response given a new predictor trajectory.

The rest of this chapter is structured as follows. Section 4.2 details the methods used for FPCA based on local polynomial regression with a real data application. FPCA-based FLR is described in Sect. 4.3 with an example when both the response and the predictor are functions. Section 4.4 includes

selective topics in FDA which have applications in engineering and other fields. All the analyses in Sects. 4.2 and 4.3 are implemented using the same public available statistical software package, and the details of the implementation are also provided. Some other statistical software packages for FDA are described in Sect. 4.4.

4.2 Functional Principal Component Analysis

4.2.1 Model for FPCA

In FDA, it is often assumed that data are independent realizations of a smooth stochastic process, $X(t)$, over a compact interval. Without loss of generality, we assume the interval to be $[0, 1]$. In this chapter, we refer t as a time variable, but it could be other index variables, such as locations in image analysis. The random process $X(t)$ is often assumed to belong to a Hilbert space, $L^2([0, 1])$, with finite second moment; that is, $E\{\int_0^1 X^2(t)dt\} < \infty$. The mean function of $X(t)$ is defined as $\mu(t) = E\{X(t)\}$, and the covariance function is denoted by $C(s, t) = \text{Cov}(X(s), X(t))$.

In real-world applications, we never observe the whole process over the time interval. Instead, data are collected discretely. There are roughly two typical types of designs depending on how the data are collected. Assume that there are n subjects. For each subject $i = 1, \dots, n$, the observations are made at time $T_{ij}, j = 1, \dots, N_i$ of the process $X_i(t)$, where $X_i(t)$ are independent realizations of $X(t)$. If the number of observations per subject, N_i , is not bounded and the observations are made on a regular grid, then it is called dense design; if N_i is bounded and observations are made randomly during the time interval, it is called sparse design.

The data set we will use for the illustration of FPCA is a bike-sharing data set. This data set contains hourly count of rental bikes during 2011 and 2012 in the Capital Bikeshare system with weather and seasonal information. After removing an obvious outlier curve, we chose days in 2011 which are neither a weekend nor a holiday. There are 249 days in total. In this section, we are interested in the count of rental bikes observed hourly as shown in Fig. 4.4. As we expected, there are two peaks of those curves: one is in the morning when people go to work, and the other is in the afternoon when people go home after work. We should mention that the design of this data might neither be a dense nor sparse design because for some days there are less than 24 observations. The method we described in the section is based on [4] which is originally proposed to deal with sparse functional data. However, it can still be applied to such types of functional data or dense functional data.

To represent the variation in functional data, FPCA has become one of the most popular methods because

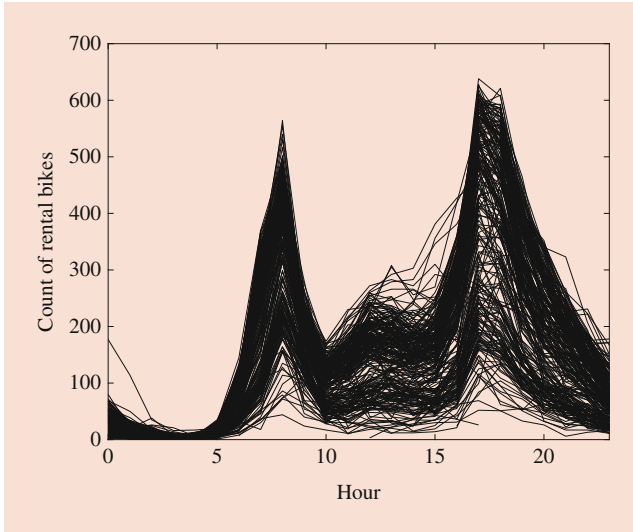


Fig. 4.4 The count of rental bikes measured hourly for 249 days in 2011. Each curve represents observations during a day

it can explain most of the variance in the data with a small number of components. According to Karhunen-Loève theorem, each random curve can be expressed as $X_i(t) = \mu(t) + \sum_{k=1}^{\infty} \xi_{ik} \phi_k(t)$, where ξ_{ik} , called PC scores, are uncorrelated zero-mean random variables with variance λ_k such that $\sum_{k=1}^{\infty} \lambda_k < \infty$. Most of the time, the measurements are contaminated with noise. For example, miscounting could happen in the Capital Bikeshare system. To incorporate possible measurement errors, assume that Y_{ij} , realizations of the process $Y(t) = X(t) + \epsilon$, are the observations we have and can be written as $Y_{ij} = \mu(T_{ij}) + X_i(T_{ij}) + \epsilon_{ij}$, where ϵ_{ij} are zero-mean measurement errors with constant variance σ^2 . The errors ϵ_{ij} are assumed to be independent of PC scores. The whole model can be expressed as

$$Y_{ij} = \mu(T_{ij}) + \sum_{k=1}^{\infty} \xi_{ik} \phi_k(T_{ij}) + \epsilon_{ij}. \quad (4.1)$$

4.2.2 Estimation of Mean Function, Eigenvalues, and Eigenfunctions

We first estimate the mean function $\mu(t)$. In FDA, it is often assumed that the functions in (4.1) are smooth. Therefore, smoothing methods can be applied to estimate unknown functions or surfaces. Yao et al. [4] used local polynomial [5] smoothing method, and [6, 7] employed spline-based approaches. Here, we introduce the local polynomial-based approach described by Yao et al. [4]. At time point t , the value of the mean function can be estimated by minimizing

$$\sum_{i=1}^n \sum_{j=1}^{N_i} \{Y_{ij} - a_0 - a_1(T_{ij} - t)\}^2 K\{(T_{ij} - t)/h_{\mu}\}$$

with respect to (a_0, a_1) , where $K(\cdot)$ is a symmetric density function, called a kernel function, and h_{μ} is a smoothing parameter, called a bandwidth, which controls the smoothness of the resulting function. Explicit solutions can be obtained and are denoted by (\hat{a}_0, \hat{a}_1) . The estimator of the mean function is $\hat{\mu}(t) = \hat{a}_0$. Rice, and Silverman [6] proposed a cross-validation method to choose the bandwidth, h_{μ} .

We now apply the local linear smoother to the bike-sharing data to obtain the mean function estimate. The package we use to implement the data analysis in this chapter is called PACE package. PACE is written in Matlab and provides functions to implement various methods in FDA. Here is the link to PACE: <http://www.stat.ucdavis.edu/PACE/>. The function to perform FPCA is also called FPCA in PACE. We applied this function to the data and obtained the estimated mean function in Fig. 4.5. The estimated curve clearly shows the pattern during a regular working day: the number of sharing bikes increases during an early morning until a smaller peak around 8 AM; then it decreases until noon and increases again; it reaches the second and bigger peak around 6 PM and then decreases again.

Define the residuals as $e_{ij} = Y_{ij} - \hat{\mu}(T_{ij})$. Similarly, the covariance function can also be estimated using local linear regression. At time point (s, t) , the estimated covariance function can be obtained by minimizing

$$\sum_{i=1}^n \sum_{j \neq k} \{e_{ij} e_{ik} - b_0 - b_1(T_{ij} - s) - b_2(T_{ik} - t)\}^2 \times K\{(T_{ij} - s)/h_C\} K\{(T_{ik} - t)/h_C\},$$

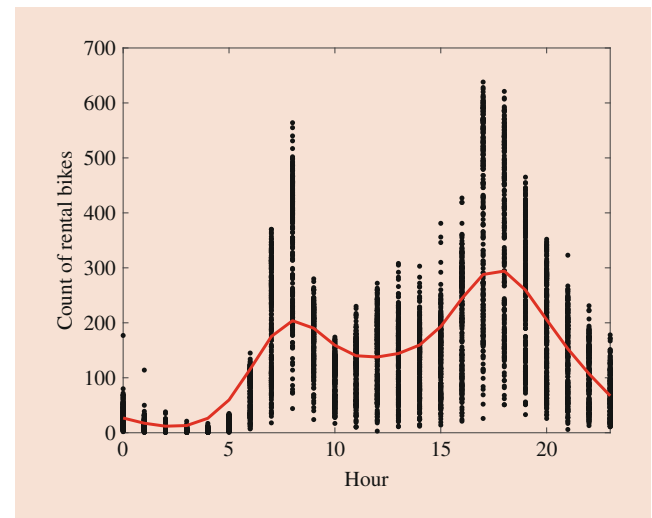


Fig. 4.5 The scatter plot of the count of the bike-sharing data and the estimated mean function using local linear smoothing

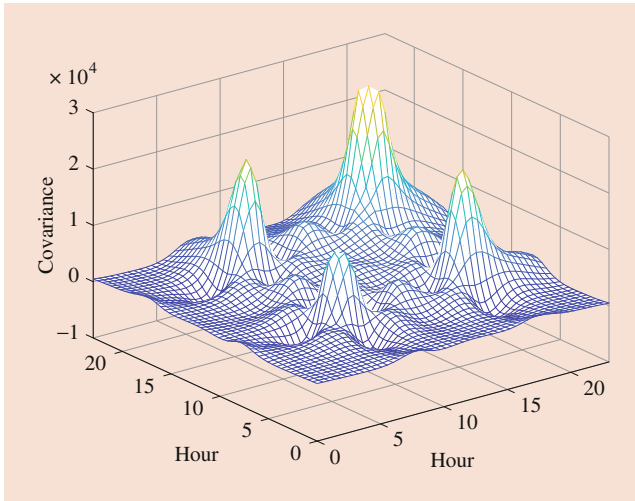


Fig. 4.6 The estimated covariance function using a local linear smoother

where h_C is a bandwidth. Denote the solutions as $(\widehat{b}_0, \widehat{b}_1, \widehat{b}_2)$; then the estimate of the covariance function is $\widehat{C}(s, t) = \widehat{b}_0$. We applied the FPCA function of the PACE package to the real data and obtained the estimated covariance function in Fig. 4.6. The main diagonal of the estimated covariance surface has two peaks located around 8 AM and 6 PM, which is consistent with the scatter plot in Fig. 4.5. We also see that the covariance between the two time points of the two peaks is large. One explanation for this phenomenon is that many people using rental bikes to go to work in the morning end up using them to go back home as well.

Let $V(t) = \text{Var}\{Y(t)\} = C(t, t) + \sigma^2$. Similarly, we estimate $V(t)$ using local linear regression and denote the solution as $\widehat{V}(t)$. The error variance σ^2 is estimated by $\widehat{\sigma}^2 = \int_0^1 \{\widehat{V}(t) - \widehat{C}(t, t)\} dt$. For the bike-sharing data, the estimated standard deviation is $\widehat{\sigma} = 53.51$, which indicates a significant amount of measurement errors.

The Karhunen-Loève theorem leads to $C(s, t) = \sum_{k=1}^{\infty} \lambda_k \phi_k(s) \phi_k(t)$. Therefore, the estimated eigenvalues, $\widehat{\lambda}_k$, and eigenfunctions, $\widehat{\phi}_k(t)$, are estimated by the following equations:

$$\int_0^1 \widehat{C}(s, t) \widehat{\phi}_k(s) ds = \widehat{\lambda}_k \widehat{\phi}_k(t), \quad (4.2)$$

where the eigenfunctions are orthonormal which means $\int_0^1 \widehat{\phi}_k(t) \widehat{\phi}_l(t) dt = \mathbf{I}(k = l)$. In practice, the eigenfunctions are often estimated by discretizing the estimated smooth covariance surface [4, 6].

Notice that the Karhunen-Loève theorem leads to an infinite sum. In real data applications, it is often truncated to a relatively large number first, e.g., 25. A criterion can then be used to select first few dominant components. A

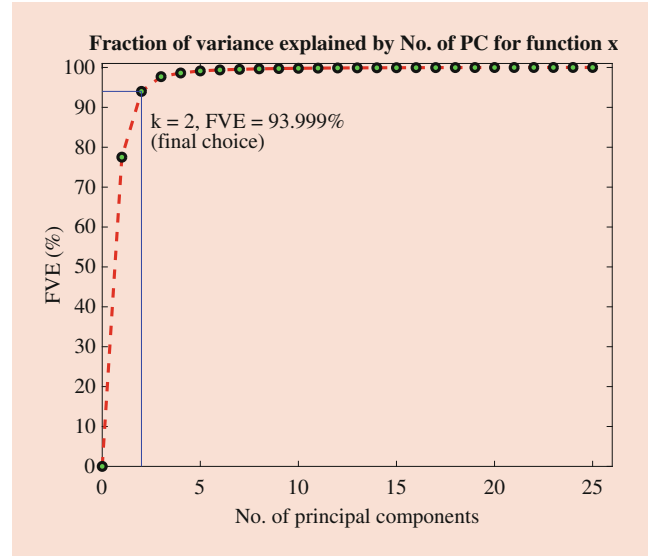


Fig. 4.7 The scree plot for FPCA of the bike-sharing data using 90% as a threshold

simple criterion is the fraction of variance explained (FVE) method. This method needs us to specify a threshold fraction first, such as 90%. If the first K components can explain at least 90% of the total variation, then FVE will choose the first K components. For the application to the bike-sharing data, if we set 90% as the threshold, the FVE method chooses the first two components as shown in the scree plot, Fig. 4.7. FVE method is simple, but a drawback of it is it is a subjective method. Data-driven approaches, such as AIC and BIC proposed in [8], are better approaches to choose the dominant components in FPCA. The first two estimated eigenfunctions are shown in Figs. 4.8 and 4.9. The other way to visualize the effects is to plot components as perturbation of the mean. Ramsay and Silverman [1] suggests to use $\widehat{\mu}(t) \pm 0.2c\widehat{\phi}_k(t)$, where $c = \sqrt{\|\widehat{\mu}(t) - \bar{\mu}\|^2}$ and $\bar{\mu} = \int_0^1 \widehat{\mu}(t) dt$. The perturbation plot for the first component is shown in Fig. 4.10. It shows that the effect of the first PC of variation is approximately adding or subtracting larger values when the value of the estimated mean function is larger. On the other hand, the effect of the second PC of variation, as shown in Fig. 4.11, shows an opposite pattern. It adds or subtracts smaller values when the value of the estimated mean function is larger.

4.2.3 Estimation of PC Scores

The next step is to estimate the PC scores $\xi_{ik} = \int_0^1 \{X_i(t) - \mu(t)\} \phi_k(t) dt$. For dense functional data, numerical integration approximation is suitable. However, for sparse functional data when there are only a few observations per subject,

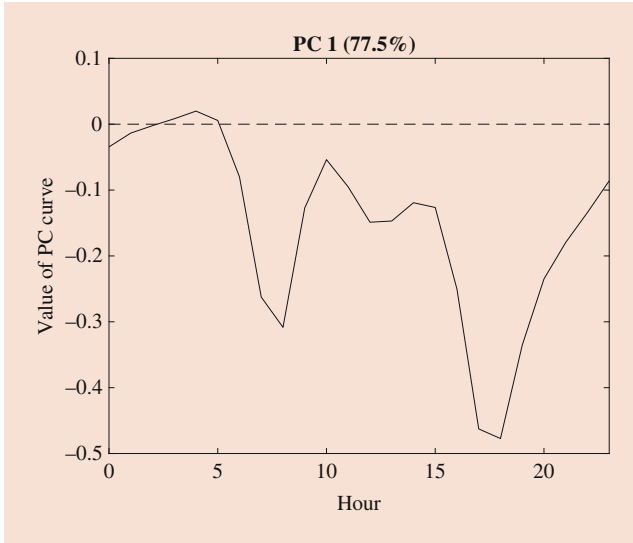


Fig. 4.8 The first eigenfunction obtained from the FPCA of the bike-sharing data

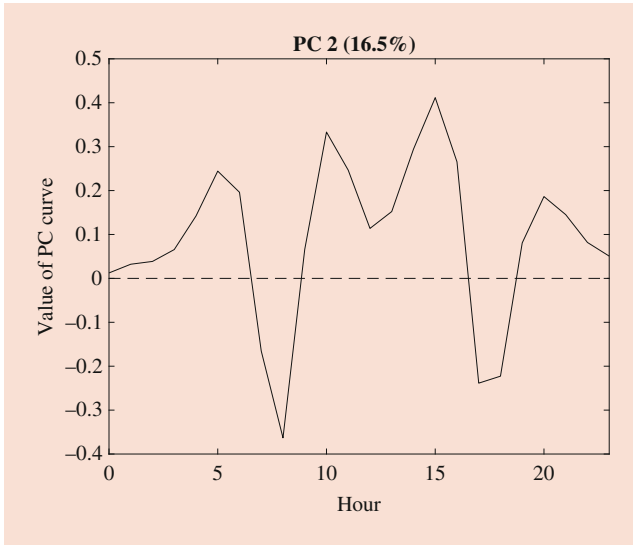


Fig. 4.9 The second eigenfunction obtained from the FPCA of the bike-sharing data

numerical integration approximation is very inaccurate. One major contribution of [4] is they proposed an algorithm called the Principal Analysis by Conditional Estimation (PACE) to solve the difficulty for sparse functional data. The advantage of PACE is it pool time points from the entire sample to predict the PC scores. Define $\tilde{X}_i = (X_i(T_{i1}), \dots, X_i(T_{iN_i}))'$, $\tilde{Y}_i = (Y_{i1}, \dots, Y_{iN_i})'$, $\mu_i = (\mu(T_{i1}), \dots, \mu(T_{iN_i}))'$, and $\phi_{ik} = (\phi_k(T_{i1}), \dots, \phi_k(T_{iN_i}))'$. Then, model (4.1) becomes a linear mixed model when the infinite sum is truncated to a finite number, K (e.g., for the bike-sharing data, $K = 2$). Assuming that the PC scores and measurement errors are joint normal, best linear unbiased prediction (BLUP) can be used to predict the functional PC scores. Specifically,

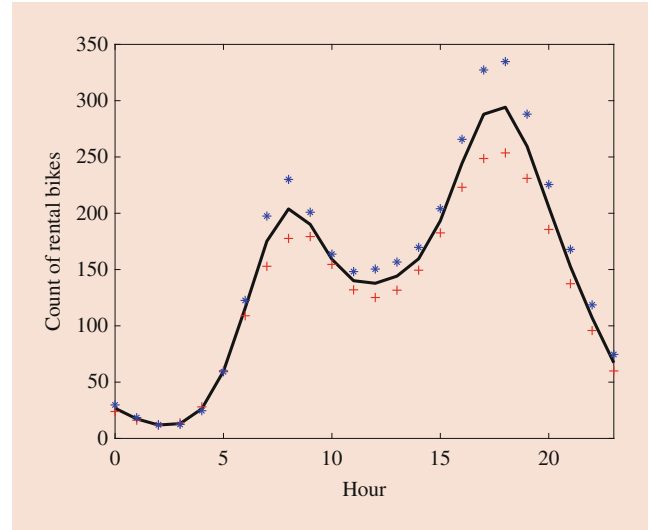


Fig. 4.10 The mean curve and the effects of adding (+) and subtracting (*) a multiple of the first eigenfunction

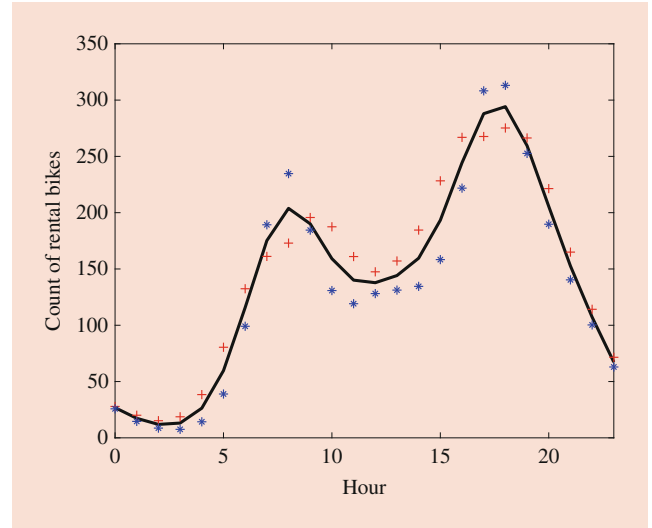


Fig. 4.11 The mean curve and the effects of adding (+) and subtracting (*) a multiple of the second eigenfunction

$$\hat{\xi}_{ik} = \hat{\lambda}_k \hat{\phi}_{ik}' \hat{\Sigma}_{Y_i}^{-1} (\tilde{Y}_i - \hat{\mu}_i), \quad (4.3)$$

where the (l_1, l_2) th element of $\hat{\Sigma}_{Y_i}$ is $\hat{C}(T_{il_1}, T_{il_2}) + \hat{\sigma}^2 \mathbf{I}(l_1 = l_2)$. According to the properties of BLUP, regardless of whether the Gaussian assumption holds or not, $\hat{\xi}_{ik}$ is the best linear prediction of ξ_{ik} .

4.2.4 Prediction of Trajectories and Confidence Bands

Now, the model (4.1) has been fitted and truncated to a suitable number of PC. The fitted model can then be used

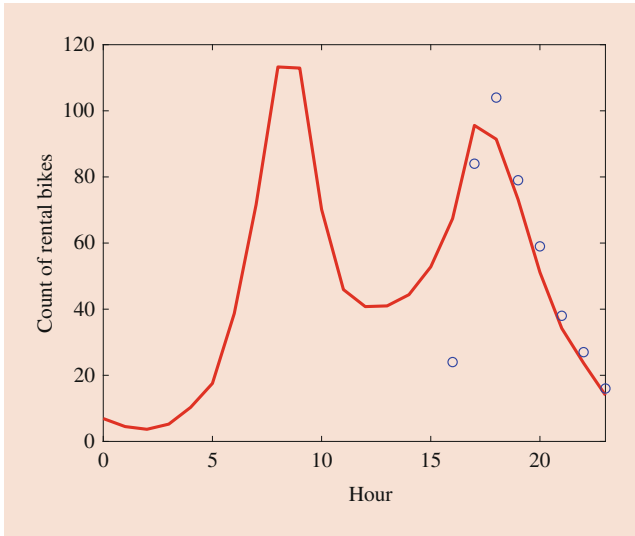


Fig. 4.12 The predicted number of rental bikes for the 18th day of the bike-sharing data. The observed values are marked as circles

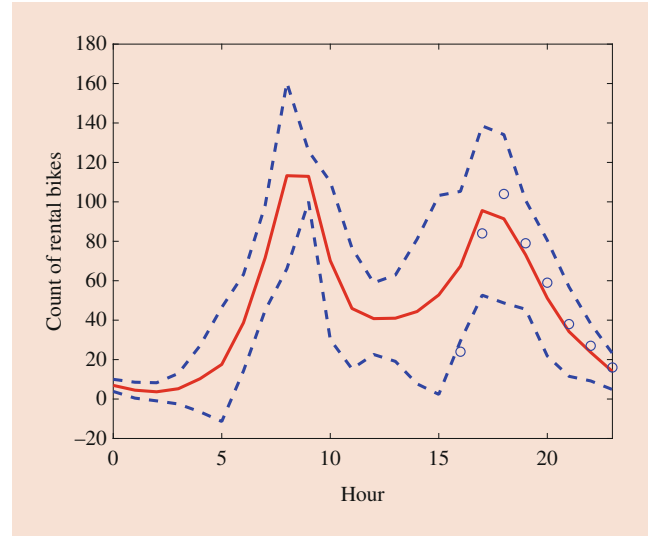


Fig. 4.13 The predicted number of rental bikes for the 18th day of the bike-sharing data and its 95% point-wise confidence band. The observed values are marked as circles

to predict the trajectory $X_i(t)$ for the i th subject. Naturally, $X_i(t)$ is estimated by

$$\widehat{X}_i(t) = \widehat{\mu}(t) + \sum_{k=1}^K \widehat{\xi}_{ik} \widehat{\phi}_k(t). \quad (4.4)$$

The prediction formula is more attractive in the sparse design in FDA because even if we observe only one or two observations for a certain subject, we can still recover the whole curve for this subject using the above formula because PACE can pool the information from curves of all subjects to make the prediction. For the bike-sharing data, most days there are 23 or 24 observations per curve. For illustration, we found one day there are only eight observations, from the 16th hour to the 23rd hour, as shown in Fig. 4.12. Using the prediction formula (4.4), we are able to recover the whole curve. From the predicted curve, we notice that this is actually a particular day because the first peak in the morning is taller than the second peak in the afternoon.

Further, confidence bands can also be obtained for each individual trajectory. Let $\widehat{\Omega}_K = \widehat{\Lambda} - \widehat{H} \widehat{\Sigma}_Y^{-1} \widehat{H}'$, where $\widehat{\Lambda} = \text{diag}\{\widehat{\lambda}_1, \dots, \widehat{\lambda}_K\}$, and $\widehat{H} = (\widehat{\lambda}_1 \widehat{\phi}_{i1}, \dots, \widehat{\lambda}_K \widehat{\phi}_{iK})'$. The $100(1 - \alpha)\%$ point-wise asymptotic confidence band for $X_i(t)$ is

$$\widehat{X}_i(t) \pm z_{1-\frac{\alpha}{2}} \sqrt{\widehat{\phi}_K' \widehat{\Omega}_K \widehat{\phi}_K}, \quad (4.5)$$

where $z_{1-\frac{\alpha}{2}}$ is the $(1 - \frac{\alpha}{2})$ quantile of the standard normal distribution and $\widehat{\phi}_K = (\widehat{\phi}_1(t), \dots, \widehat{\phi}_K(t))'$. Using the formula (4.5), we constructed the 95% point-wise confidence band for the predicted curve in Fig. 4.12 using the PACE package.

Table 4.1 First ten rows of the bike-sharing data

Hour	Humidity	Wind speed	Count	Day
0	0.44	0.3582	5	1
1	0.44	0.4179	2	1
4	0.47	0.3881	1	1
5	0.47	0.2836	3	1
6	0.50	0.3881	30	1
7	0.50	0.1940	64	1
8	0.50	0.2836	154	1
9	0.43	0.3881	88	1
10	0.43	0.2537	44	1
11	0.40	0.3284	51	1

The confidence band is shown in Fig. 4.13. It also supports the finding that the first peak is higher.

4.2.5 Implementation Details

In this section, we first describe the data we used for the analysis in both Sects. 4.2 and 4.3. The first ten observations of the whole data set are shown in Table 4.1. The first column represents the hour of measurements, ranging from 0 to 23. Humidity and wind speed are shown in the second and third columns. Both variables are normalized with a maximum at 1. The fourth column shows the number of total rental bikes including both casual and registered. The last column is simply a label of the day in the year 2011. More details about this data set can be found at “<https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset>.”

Below are Matlab codes used to implement the FPCA, predict, and obtain relevant figures in this section.

```

% add the toolbox to the working path
addpath(genpath('C:/Program Files/MATLAB/toolbox/release2.17/'))

% Load the data set and create 4 cells to be used
data = csvread('data.csv');      % a 5911 by 5 matrix
ncohort = 249;                    % 249 days in total
t = cell(1,ncohort);
humid = cell(1,ncohort);
wind = cell(1,ncohort);
y = cell(1,ncohort);
for i=1:ncohort
    day_i = data(data(:,5)==i,1:4);
    t{i} = day_i(:,1)';           % hour: 0 to 23
    y{i} = day_i(:,4)';          % count of bicycles used
end

% Plot the raw data of y v.s. t
figure;
for i=1:ncohort
    plot(t{i},y{i},'k');
    hold on;
end
xlim([0 23]);
xlabel('Hour');
ylabel('Count of rental bikes');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Conduct FPCA and obtain results %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
regular = 1; %Case iii) regular data with missing values (regular = 1)
p = setOptions('yname','x', 'regular', regular, 'selection_k', 'FVE',
    'FVE_threshold',0.9 , 'screePlot',1,'numBins',0, 'verbose','on');
[result_fpca] = FPCA(y,t,p);
out1 = getVal(result_fpca,'out1'); %vector of time points for mu, phi and
    %ypred
mu = getVal(result_fpca,'mu'); %estimated mean function
out21 = getVal(result_fpca,'out21'); %vector of time points for xcov
xcov = getVal(result_fpca,'xcov'); %estimated smooth covariance evaluated
    %at out21
xcorr = getVal(result_fpca,'xcorr'); %fitted correlation surface
phi = getVal(result_fpca,'phi'); %estimated eigenfunctions
FVE = getVal(result_fpca,'FVE'); %fraction of variance explained
sigma = getVal(result_fpca,'sigma'); %estimate of measurement error variance
xi_var = getVal(result_fpca,'xi_var'); %a cell of K*K matrices, omega matrix in
    %equation (7) of the JASA paper

```

```
% Plot the estimated mean function
figure;
plot(cell2mat(t),cell2mat(y),'k.','MarkerSize',10);
hold on;
plot(out1,mu,'r','LineWidth',2);
xlim([0 23]);
xlabel('Hour');
ylabel('Count of rental bikes');

% Plot the estimated covariance surface
figure;
mesh(out21,out21,xcov);
xlabel('Hour');
ylabel('Hour');
zlabel('Covariance');
xlim([0 23]);
ylim([0 23]);

% Plot the estimated correlation surface
figure;
mesh(out21,out21,xcorr);
xlabel('Hour');
ylabel('Hour');
zlabel('Correlation');
xlim([0 23]);
ylim([0 23]);

% Plot the estimated eigen-functions
figure;
plot(out1,phi(:,1),'k');
hold on;
plot(out1,zeros(1,length(out1)),'k--');
xlim([0 23]);
xlabel('Hour');
ylabel('Value of PC curve');
title('PC 1 (77.5%)');

figure;
plot(out1,phi(:,2),'k');
hold on;
plot(out1,zeros(1,length(out1)),'k--');
xlim([0 23]);
xlabel('Hour');
ylabel('Value of PC curve');
title('PC 2 (16.5%)');
```

```

% Plot the effects plot
mu_bar = mean(mu);
C = sqrt(mean((mu-mu_bar).^2));
figure;
plot(out1,mu,'k','LineWidth',2);
hold on;
plot(out1,mu+C*phi(:,1),'r+','MarkerSize',5);
hold on;
plot(out1,mu-C*phi(:,1),'b*','MarkerSize',5);
xlim([0 23]);
xlabel('Hour');
ylabel('Count of rental bikes');

figure;
plot(out1,mu,'k','LineWidth',2);
hold on;
plot(out1,mu+C*phi(:,2),'r+','MarkerSize',5);
hold on;
plot(out1,mu-C*phi(:,2),'b*','MarkerSize',5);
xlim([0 23]);
xlabel('Hour');
ylabel('Count of rental bikes');

% Prediction for the 18-th cell of t
i = 18;
ypred=FPCAeval(result_fpca,[],out1); %obtain predicted curves for all existing
    subjects
figure;
plot(out1,ypred{i},'r','LineWidth',2);
hold on;
plot(t{i},y{i},'bo');
xlim([0 23]);
xlabel('Hour');
ylabel('Count of rental bikes');

% Point-wise C.I. for X_i(t)
sd = sqrt(diag(phi*xi_var{i}*phi'));
figure;
plot(out1,ypred{i},'r','LineWidth',2);
hold on;
plot(t{i},y{i},'bo');
hold on;
plot(out1,ypred{i}+1.96*sd,'b--','LineWidth',2);
hold on;
plot(out1,ypred{i}-1.96*sd,'b--','LineWidth',2);
xlim([0 23]);
xlabel('Hour');
ylabel('Count of rental bikes');

```


4.3 Functional Linear Regression

4.3.1 Model for FLR

A common case in FLR is both a response variable and predictors are functions over compact sets. For example, in the bike-sharing data set, one predictor, which might be useful to predict the number of rental bikes, is humidity measurement over time. In this section, we mainly focus on the situation when there is a continuous covariate measured over a compact set. The methodology is proposed in [9] for sparse functional data regression, but it may also be applied to other designs of functional data.

Without loss of generality, we also assume the compact set to be $[0,1]$. Assume that the observed values of the predictor are W_{il} , $i = 1, \dots, n$, $l = 1, \dots, M_i$. Similar to (4.1), we write the model for the predictor as

$$\begin{aligned} W_{il} &= V_i(S_{il}) + \varepsilon_{il} \\ &= \mu_V(S_{il}) + \sum_{k=1}^{\infty} \zeta_{ik} \psi_k(S_{il}) + \varepsilon_{il}, \end{aligned} \quad (4.6)$$

where $\mu_V(s) = E\{V_i(s)\}$ is the mean function, S_{il} are measurement times for V_i , ζ_{ik} are zero-mean functional PC scores with variance ρ_k , $\psi_k(s)$ are eigenfunctions, and ε_{il} are zero-mean measurement errors with variance σ_V^2 . The same FPCA procedure described in Sect. 4.2 can be applied to the predictor trajectory to obtain estimates in model (4.6).

We now apply FPCA to the predictor in the bike-sharing data. To make predictions later, we only use the first 247 days for model fitting, and the remaining last 2 days are used for functional prediction. The humidity variable considered here is normalized by dividing it by the maximum, 100. The trajectories of the normalized humidity and the estimated mean function are shown in Fig. 4.14. Notice that the maximum humidity may appear any time during a day because of rain. The smooth estimate, $\hat{v}(s)$, shows a sinusoidal pattern, reaching a maximum around 5 AM when the temperature is about the lowest during a day and reaching a minimum around 3 PM when the temperature is close to the highest during a day. We set 90% as the threshold, and the FVE method chooses the first three PC. The estimated first three eigenfunctions are shown in Fig. 4.15. The first eigenfunction is close to a constant and reflects an overall level of shift. The second eigenfunction represents a contrast between early and late times during a day, and the third eigenfunction shows a contrast between the middle times and the times of the two tails. In this application, both humidity and the count of rental bikes are measured hourly. However, this is not required in FLR. A covariate function could be measured over a different compact set.

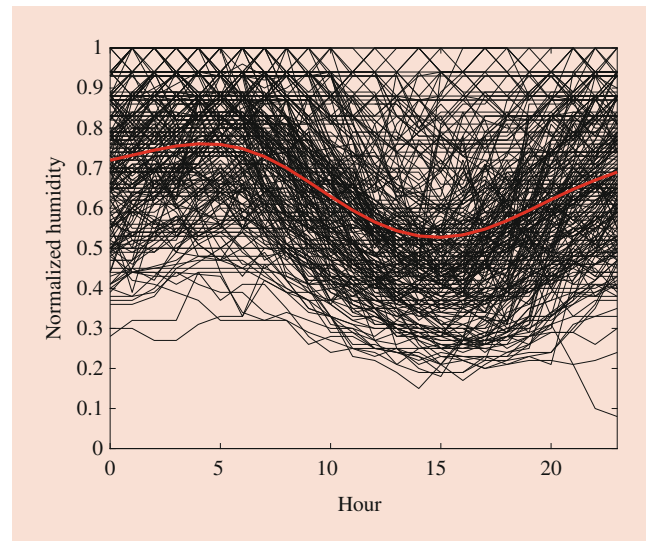


Fig. 4.14 Observed curves of normalized humidity in the bike-sharing data and the estimated mean function using local linear smoothing

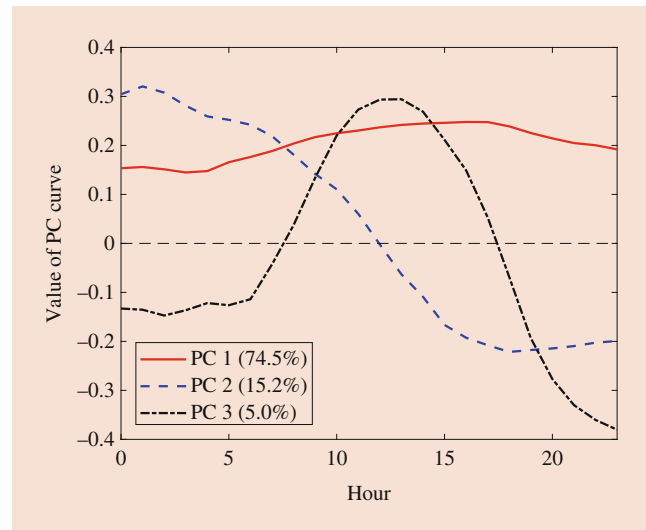


Fig. 4.15 Smooth estimates of the first (solid), second (dashed), and third (dash-dot) eigenfunctions for the normalized humidity in the bike-sharing data

A FLR model when both the predictor and response are random functions can be expressed by

$$E[X(t)|V] = \alpha(t) + \int_0^1 \beta(s, t) V(s) ds, \quad (4.7)$$

where $\beta(s, t)$ is a smooth and square integrable bivariate regression function. The above model (4.7) is a natural generalization of the simple linear regression model, $E[X|V] = \alpha + \beta V$. It is well known that for a simple linear regression model, $\beta = \text{Cov}(V, X)/\text{Var}(V)$. For FLR, under certain

regularity conditions, this representation still holds. The regression function is given by

$$\begin{aligned}\beta(s, t) &= \sum_{k=1}^{\infty} \sum_{m=1}^{\infty} \frac{\text{Cov}(\zeta_m \psi_m(s), \xi_k \phi_k(t))}{\|\text{Var}(\zeta_m \psi_m(s))\|^2} \\ &= \sum_{k=1}^{\infty} \sum_{m=1}^{\infty} \frac{\text{E}[\zeta_m \xi_k]}{\text{E}[\zeta_m^2]} \psi_m(s) \phi_k(t).\end{aligned}\quad (4.8)$$

4.3.2 Estimation of Regression Function and R^2

To estimate $\beta(s, t)$ in (4.8), we need to obtain the estimators of the covariance among the PC scores, $\sigma_{km} = \text{E}[\zeta_m \xi_k]$. By the Karhunen-Loève theorem of the predictor and response processes, it follows that the cross-covariance surface between the two processes is

$$\text{Cov}(V(s), X(t)) = \sum_{k=1}^{\infty} \sum_{m=1}^{\infty} \text{E}[\zeta_k \xi_m] \psi_k(s) \phi_m(t).$$

Similarly, the cross-covariance surface can be estimated using local polynomial smoothing by smoothing the ‘‘raw’’ cross-covariances, $\{W_{ij} - \widehat{\mu}_V(S_{ij})\}\{Y_{ij} - \widehat{\mu}(T_{ij})\}$. Let $\widehat{\text{Cov}}(V(s), X(t))$ be the estimated cross-covariance surface; then, $\sigma_{km} = \text{E}[\zeta_m \xi_k]$ is obtained by

$$\int_0^1 \int_0^1 \widehat{\psi}_m(s) \widehat{\text{Cov}}(V(s), X(t)) \widehat{\phi}_k(t) ds dt.$$

Assume that the estimated numbers of PC for the predictor and response process are M and K separately. The estimator for $\beta(s, t)$ becomes

$$\widehat{\beta}(s, t) = \sum_{k=1}^K \sum_{m=1}^M \frac{\widehat{\sigma}_{km}}{\widehat{\rho}_m} \widehat{\psi}_m(s) \widehat{\phi}_k(t).\quad (4.9)$$

We now apply the procedure to the bike-sharing data. The function which performs the above procedure is FPCReg. Previous FPCA for the predictor and the response has shown that $M = 3$ and $K = 2$ (the same as we expected when only the last 2 days are dropped). Using (4.9), we obtained the estimated regression function in Fig. 4.16. The contour plot clearly shows that during the first 5 hours of a day, the humidity does not contribute to the prediction of the number of rental bikes. The regression function seems to have two big peaks, two small peaks, and two valleys. Interestingly, they all located around the time when the time of humidity is around 9 AM and 6 PM.

In regression models, the coefficient of determination is often used to measure how well the observed responses are

predicted by the model. It reflects the proportion of the variance in the dependent variable that is predictable from the predictors. In [9], a functional counterpart of the coefficient of determination is also proposed. A global coefficient of determination, denoted by R^2 , is defined as

$$\begin{aligned}R^2 &= \int_0^1 \frac{\text{Var}(\text{E}[X(t)|V])}{\text{Var}(X(t))} dt \\ &= \int_0^1 \frac{\sum_{m=1}^{\infty} \sum_{k,l=1}^{\infty} \sigma_{km} \sigma_{lm} \phi_k(t) \phi_l(t) / \rho_m}{\sum_{k=1}^{\infty} \lambda_k \phi_k^2(t)} dt.\end{aligned}$$

A natural estimator of R^2 is

$$\widehat{R}^2 = \int_0^1 \frac{\sum_{m=1}^M \sum_{k,l=1}^K \widehat{\sigma}_{km} \widehat{\sigma}_{lm} \widehat{\phi}_k(t) \widehat{\phi}_l(t) / \widehat{\rho}_m}{\sum_{k=1}^K \widehat{\lambda}_k \widehat{\phi}_k^2(t)} dt.\quad (4.10)$$

Applying the formula (4.10) to the real data, we got $\widehat{R}^2 = 0.14$, which indicates that humidity does not contribute very much to the prediction of the number of rental bikes. One may be wondering whether the contribution is significant or not. The Rtest function in PACE allows a bootstrapped sampling-based method to test whether the R-square is zero or not. We set the number of bootstrapped samples to be 500 and obtained a P-value smaller than 0.002, which states that the correlation is significant.

4.3.3 Prediction and Confidence Bands

One of the most important roles of a regression model is it allows us to predict the response for a new subject given a predictor of the subject. In functional regression, this can also be performed similarly. By the properties of conditional expectation, Eq. (4.7) has the following equivalent expression:

$$\text{E}[X(t)|V] = \mu(t) + \int_0^1 \beta(s, t) \{V(s) - \mu_V(s)\} ds.\quad (4.11)$$

Given a predictor process V^* , (4.8), (4.11), and the orthogonality of eigenfunctions leads to

$$\text{E}[X^*(t)|V^*] = \mu(t) + \sum_{k=1}^{\infty} \sum_{m=1}^{\infty} \frac{\sigma_{km}}{\rho_m} \zeta_m^* \phi_k(t),\quad (4.12)$$

where $\zeta_m^* = \int_0^1 \{V^*(s) - \mu_V(s)\} \psi(s) ds$. Except ζ_m^* , all quantities in (4.12) have been obtained in the model fitting. The PC scores of V^* , ζ_m^* , may be estimated easily for dense functional data by the plug-in of V^* , $\widehat{\mu}_V(s)$, and $\widehat{\psi}(s)$. For sparse functional data design, ζ_m^* can be estimated through BLUP (see [9]) for details). After the number of PC has been determined, the predicted trajectory, X^* , is

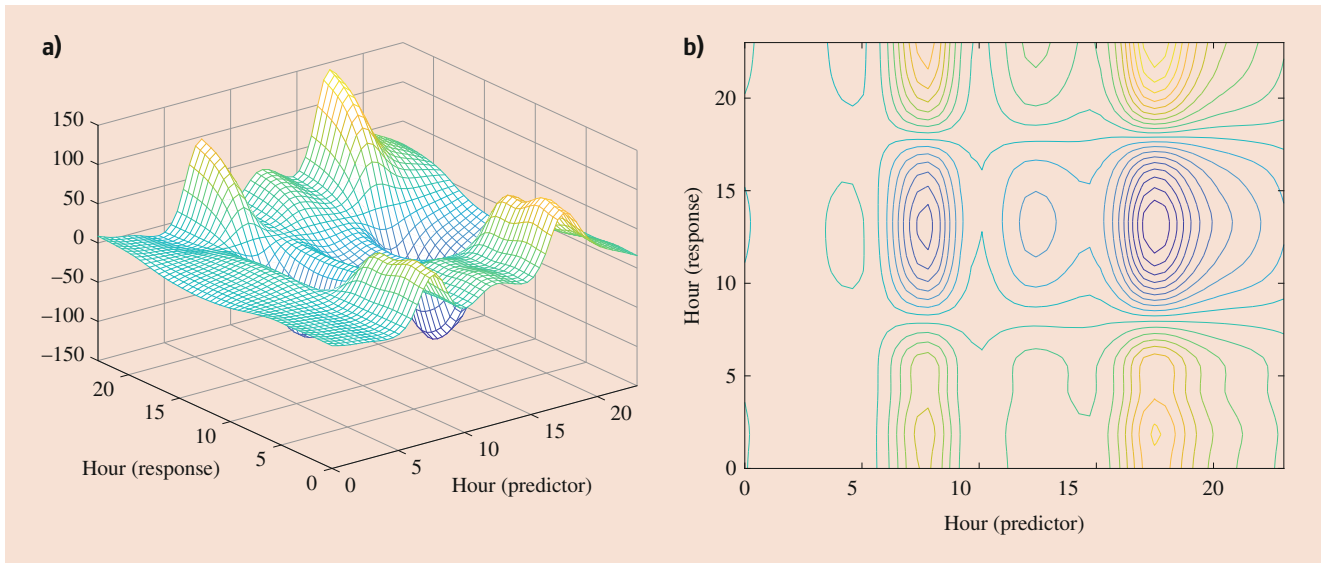


Fig. 4.16 Estimated regression function when the predictor is normalized humidity and the response is number of rental bikes for the bike-sharing data. (a) is the 3-D plot of the estimated regression function; (b) is the corresponding contour plot

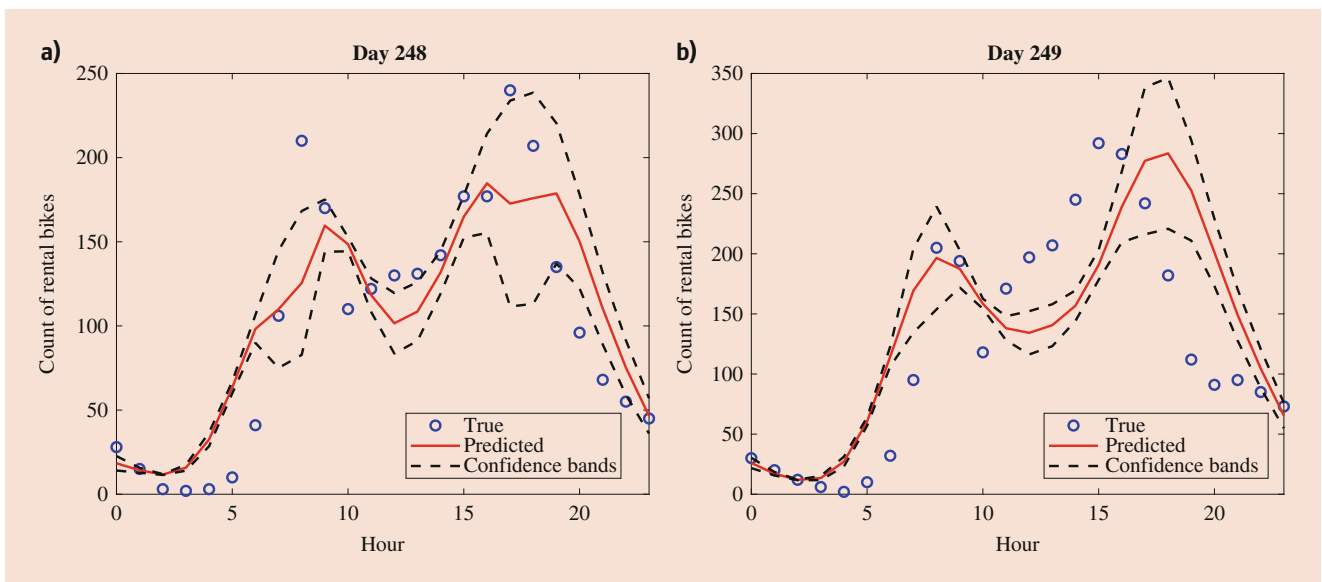


Fig. 4.17 Two days of observed values (circles) of bike-sharing data that are not used for the prediction. The predicted curves (solid) and corresponding 95% point-wise confidence bands (dashed), for the last 2 days in the data. (a) prediction for the 248th day; (b) prediction for the 249th day

$$\widehat{X}^*(t) = \widehat{\mu}(t) + \sum_{k=1}^K \sum_{m=1}^M \frac{\widehat{\sigma}_{km}}{\widehat{\rho}_m} \widehat{\zeta}_m^* \widehat{\phi}_k(t). \quad (4.13)$$

The asymptotic point-wise confidence bands for $X^*(t)$ are also provided in [9].

We applied the procedures to the bike-sharing data to make predictions of the number of rental bike trajectories of the last 2 days using the data of the previous days. The estimated trajectories and their corresponding 95% confidence bands are shown in Fig. 4.17. The prediction is not bad

considering that humidity is not a very important variable to predict the number of rental bikes and only contributes 14% to the explanation of the variance.

4.3.4 Implementation Details

The description of the data is in Sect. 4.2.5. Below are Matlab codes used to implement the functional regression, predict, and obtain relevant figures in this section.

```

% add the toolbox to the working path
addpath(genpath('C:/Program Files/MATLAB/toolbox/release2.17/'))

% Load the data set and create 4 cells to be used
data = csvread('data.csv');      % a 5911 by 5 matrix
ncohort = 249;                    % 249 days in total
t = cell(1,ncohort);
humid = cell(1,ncohort);
wind = cell(1,ncohort);
y = cell(1,ncohort);
for i=1:ncohort
    day_i = data(data(:,5)==i,1:4);
    t{i} = day_i(:,1)';           % hour: 0 to 23
    humid{i} = day_i(:,2)';
    wind{i} = day_i(:,3)';
    y{i} = day_i(:,4)';          % count of bicycles used
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Conduct Functional linear regression %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% set parameter options for FPCA, default ngrid is 50, ngrid=200 for more
% smooth curve
param_X = setOptions('selection_k','FVE','FVE_threshold',0.9,'verbose','on');
param_Y = setOptions('selection_k','FVE','FVE_threshold',0.9,'verbose','on');
FIT = 0;          %Fitting a functional linear regression (default) through
                  %decomposition into
                  %simple linear regressions between the FPC scores of Y and X
K_x = [];        %Number of PC of functional predictor X used in functional
                  %regression
K_y = [];        %Number of PC of functional predictor Y used in functional
                  %regression
npred = 2;       %the last 2 subjects and their values for t_x, t_y are used for
                  %prediction
alpha = 0.05;    %the level of the confidence bands.  alpha = 0.05 if is left
                  %empty.

% fit functional linear regression using humid
% Remark: if using wind, Q=0.04 and r2=0.09, too small for prediction
[res, xx, yy] = FPCreg(humid,t,y,t,param_X,param_Y,FIT,K_x,K_y,npred,alpha);
%Functional regression

% Plot the humidity data wind v.s. t with estimated mean function
out1 = getVal(xx,'out1');         %vector of time points for mu, phi and ypred
mu = getVal(xx,'mu');            %estimated mean function
figure;
for i=1:ncohort
    plot(t{i},humid{i},'k');
    hold on;
end
hold on;
plot(out1,mu,'r','LineWidth',2);
xlim([0 23]);

```

```

xlabel('Hour');
ylabel('Normalized humidity');

% Plot the estimated eigen-functions of the FPCA of the predictor
phi = getVal(xx,'phi');           %estimated eigenfunctions
figure;
plot(out1,phi(:,1),'r-',out1,phi(:,2),'b--',out1,phi(:,3),'k-.','LineWidth',1.5);
hold on;
plot(out1,zeros(1,length(out1)),'k--');
legend('PC 1 (74.5%)','PC 2 (15.2%)','PC 3 (5.0%)','Location','SouthWest')
xlim([0 23]);
xlabel('Hour');
ylabel('Value of PC curve');

% plot the estimated beta function: 3-D plot
figure;
BETAest = getVal(res,'BETA');
mesh(getVal(BETAest,'grid_x'),getVal(BETAest,'grid_y'),getVal(BETAest,'beta'));
xlim([0 23]);
xlabel('Hour (predictor)');
ylim([0 23]);
ylabel('Hour (response)');

% plot the estimated beta function: contour plot
BETAest = getVal(res,'BETA');
figure;
[X,Y] = meshgrid(getVal(BETAest,'grid_x'),getVal(BETAest,'grid_y'));
contour(X,Y,getVal(BETAest,'beta'),20);
xlim([0 23]);
xlabel('Hour (predictor)');
ylim([0 23]);
ylabel('Hour (response)');

% plot true and predicted response curves (predict the last 2 observations)
new_true_y = y((ncohort-npred+1):ncohort); % true response curves for prediction
%part
new_ty = t((ncohort-npred+1):ncohort);      % time points corresponding to the
% newy and newEy
newy = getVal(res,'newy');                   % estimated response curves
newcb = getVal(res,'newcb');
for i = 1:2
    figure;
    plot(new_ty{i},new_true_y{i},'bo',new_ty{i},newy{i},'r-',new_ty{i},
        newcb{1,i},'k--',new_ty{i},newcb{2,i},'k--','LineWidth',1.5);
    title(['Day ' num2str(ncohort-npred+i)]);
    xlim([0 23]);
    xlabel('Hour');
    ylabel('Count of rental bikes');
    legend('True','Predicted','Confidence bands','Location','SouthEast')
end

```

```

% test whether the quasi R-square, Eq. (20) in Yao et.al. 2005, is zero or not
%based on bootstraped sampling
nsim = 500;
[rej pv] = Rtest(res, xx, yy, nsim, alpha);
%P-value < 0.002. Reject the null hypothesis of no regression relation.

% obtain functional R-square and Quasi R-square
r2 = getVal(res, 'r2'); % functional R-square, is 0.2604
Q = getVal(res, 'Q'); % Quasi R-square, is 0.1383

```

4.4 Other Selected Topics in FDA and Software Packages

In this section, we briefly review some other topics in FDA and software packages developed for FDA. The topics and packages introduced here are not intended to be comprehensive but selective. A review of recent developments in FDA is [10].

4.4.1 Other Selected Topics in FDA

Although [4] provides some asymptotic results, such as consistency of mean and covariance functions, eigenvalues and eigenfunctions, and asymptotic confidence bands, the asymptotic properties are not optimal. Hall et al. [11] further studied the asymptotic properties for the estimators proposed by Yao et al. [4]. Their work shows that the estimation of eigenvalues is a semiparametric problem, with root- n consistent estimators, even if only a few observations are made of each function and if each observation is contaminated with noise. However, the estimation of eigenfunctions becomes a nonparametric problem when observations are sparse. Hall and Vial [12] proposed techniques for assessing the finiteness of dimensionality in FDA. Li and Hsing [13] developed strong uniform convergence rates of the nonparametric estimation of the mean and covariance functions for functional/longitudinal data. Li and Hsing [14] considered regression models with a functional predictor and a scalar response. They assumed that the response depends on the predictor only through a finite number of projections and investigated the dimensionality of the linear subspace spanned by the projections. Recently, [15] provided a comprehensive analysis of the asymptotic properties of nonparametric estimation of mean and covariance functions for various types of sampling plans.

Sometimes it is of interest to recover the underlying derivative from a sample or random functions with a sparse sampling design and measurement errors. Liu and Müller [16] considered the following model for the derivative process $X_i^{(v)}(t)$:

$$X_i^{(v)}(t) = \mu^{(v)}(t) + \sum_{k=1}^{\infty} \xi_{ik} \phi_k^{(v)}(t), \quad (4.14)$$

where $\mu^{(v)}(t)$ and $\phi_k^{(v)}(t)$ denote the v th derivative of mean and eigenfunctions. It is assumed that the observations are made over irregular time points and are contaminated with errors. Liu and Müller [16] then proposed an approach based on estimating derivatives of eigenfunctions and expansions of random functions to obtain a representation for the derivative process. Dai et al. [17] studied similar problems but based on a different approach. They used a direct Karhunen-Loève expansion of the unobserved derivatives and numerically showed that the method recovers the underlying derivatives more accurately.

The FLR we described in Sect. 4.3 is called function-on-function regression, where, for each subject, both response and predictors are trajectories over compact sets. When the response is not functional but scalar, the corresponding regression model is called scalar-on-function regression. For example, for the bike-sharing data, the response could be the total number of rental bikes each day. A FLR model with scalar response can be expressed as

$$E[X|V] = \alpha + \int_0^1 \beta(t)V(t)dt, \quad (4.15)$$

where X is a scalar response, $V(t)$ is a functional covariate, and $\beta(t)$ is the coefficient function. The interpretation of such a linear regression model will be easier than the one in function-on-function regression. For a given time point t , the larger the absolute value of $\beta(t)$, the larger is the impact of V on X . For regular and dense functional data, Chapter 15 of [1] describes classical methods for scalar-on-function regression. The methods are basically estimating $\beta(t)$ by the use of basis functions, such as splines and wavelets. The main methodology of [9] can also be applied to scalar-on-function regression when covariates are sparse functional or longitudinal data. Furthermore, the PACE package also provides functions for such implementation. A review of the developments of functional regression can be found in [18]. The author highlighted the modeling structures that

have been developed in the past 10 years and the various regularization approaches employed in three types of regression models: function-on-function, scalar-on-function, and function-on-scalar, where the response is functional but the predictor is scalar.

In statistics, analysis of variance (ANOVA) is used to analyze the differences among group means in data. In FDA, when the predictor is categorical, such as group factors or treatments, and the response is a trajectory over a compact set, the corresponding model is termed as a functional ANOVA model. For example, one group factor in the bike-sharing data could be the day of a week. For regular and dense functional data, see Chapter 13 of [1] for the classical methods. Xu et al. [2] considered a recent application of functional ANOVA in plant sciences. Let $i = 1, \dots, n$ be the index of maize plant. Define genotype indicator G_i as $G_i = 1$ if the i th maize is of genotype B73 and $G_i = 0$ if the i th maize is of genotype MM. Define the environment indicator W_i as $W_i = 1$ if the i th maize is sufficiently watered and $W_i = 0$ if the i th maize is insufficiently watered. Xu et al. [2] assumed the following functional analysis of variance model for the plant growth:

$$y_i(t) = \mu(t) + G_i g(t) + W_i w(t) + \epsilon_i(t), \quad (4.16)$$

where $\mu(t)$ is the growth function of insufficiently watered MM maize, $g(t)$ is the main effect function of genotype B73, $w(t)$ is the main effect function of sufficient watering, and $\epsilon_i(t)$ is a zero-mean random process representing errors. Zhang [19] includes various major topics in functional ANOVA and some up-to-date hypothesis testing methods for FDA.

The functional data we introduced in the previous sections are actually uni-level functional data, where it is assumed that the processes are independently and identically distributed. In practice, data may have a multilevel structure. For example, [20] studied the impact of national culture on corporate environmental management, where the data have a two-level hierarchical structure because corporations are nested in countries. Di et al. [21] described a sleep heart health study, where there are two visits for each patient with functional measurements, so the data have two levels with visits nested in patients. Xu et al. [3] presented a study of the bending rates of maize roots based on FPCA of functional phenotypes collected from seeds. The data are shown in Fig. 4.3. The data have a three-level nested hierarchical structure, with seeds nested in groups nested in genotypes. Let $Y_i(t)$ be the bending rate of the i th seed at time $t \in [0, 1.5]$, $i = 1, 2, \dots, n$. Denote s_i as the lunar day on which the i th seed was measured, where $s_i \in [1, 30]$. For each seed, a covariate vector X_i is observed depending on the camera setup. Xu et al. [3] modeled the data by the following hierarchical functional data model:

$$Y_i(t) = \mu(s_i, t) + X_i' \alpha + Z_{1,g(i)}(t) + Z_{2,f(i)}(t) + Z_{3,i}(t) + \epsilon_i(t), \quad (4.17)$$

where $\mu(s_i, t)$ is the mean function of the bending rate under a baseline camera setup; α represents fixed effects of cameras; $Z_{1,g(i)}(t)$, $Z_{2,f(i)}(t)$, and $Z_{3,i}(t)$ are random processes representing the functional random effects of genotype $g(i)$, file/group $f(i)$, and seed i , respectively; and $\epsilon_i(t)$ is a white noise measurement error with variance σ^2 .

It is well known in robust statistics that principal component analysis (PCA) has the issue of non-robustness. Similarly, the issue also occurs in FDA using PCA. In FPCA, the estimation of the covariance function and the functional PC are sensible to outliers. Therefore, a robust version of FPCA is needed. Sawant et al. [22] proposed an algorithm for outlier detection and robust FPCA for FDA. The method is motivated by the combination of both projection pursuit and robust covariance estimation using the minimum covariance determinant (MCD) method. She et al. [23] introduced a method called robust orthogonal complement PC analysis. The method they proposed is called orthogonal complement principal component analysis (ROC-PCA). Their framework combines the popular sparsity-enforcing and low-rank regularization techniques together to deal with row-wise outliers and element-wise outliers. This approach can identify outliers and perform robust principal subspace recovery simultaneously.

The confidence band presented in this chapter is called a point-wise confidence band, which means that for each t , the truth is covered by the confidence interval at t with the probability $1 - \alpha$. Specifically, if the mean function to estimate is $\mu(t)$, a point-wise confidence band $\hat{\mu}(t) \pm \hat{\delta}(t)$ with coverage probability $1 - \alpha$ satisfies the following condition separately for each time point t :

$$P\{\hat{\mu}(t) - \hat{\delta}(t) < \mu(t) < \hat{\mu}(t) + \hat{\delta}(t)\} = 1 - \alpha.$$

Simultaneous confidence band, on the other hand, covers the truth for all t with the probability $1 - \alpha$. For the estimation of the mean function, that is,

$$P\{\hat{\mu}(t) - \hat{\delta}(t) < \mu(t) < \hat{\mu}(t) + \hat{\delta}(t) \text{ for all } t\} = 1 - \alpha.$$

Usually, a simultaneous confidence band will be wider than a point-wise confidence band when the coverage probability is the same. In both [4] and [9], the asymptotic confidence bands given are point-wise confidence bands. A simultaneous confidence band for the mean function in the scenario of dense functional data is proposed in [24]. The estimator is a polynomial spline estimator, and the authors showed that the confidence band has an oracle efficiency property. Cao et al. [25] further developed simultaneous confidence envelopes for the covariance function in dense functional data.

In statistics, classification is usually adopted when we want to identify to which group a new observation should be assigned given a training data set. In FDA, the counterpart is called functional classification, in which we want to assign the membership for a new functional object with a classifier. Müller [26] used a functional binary regression model for functional classification. James and Hastie [27] proposed a method to classify functions based on a functional linear discriminant analysis approach. Their technique extends the classical linear discriminant analysis method to the case where the predictor variable is functions.

Multivariate statistics is an important branch of statistics. In FDA, when many characteristics of a subject are measured over time, the data collected is, namely, multivariate functional data. Multivariate FPCA based on a normalization approach was proposed in [28]. Wong et al. [29] investigated the prediction of a scalar response by both parametric effects of a multivariate predictor and nonparametric effects of a multivariate functional predictor. To be specific, [29] model the relationship between scale response y_i , $i = 1, \dots, n$ and a predictor vector z_i and a multivariate functional predictor x_i as

$$y_i = m(z_i, x_i) + \epsilon_i,$$

where $m(\cdot, \cdot)$ is the regression function and ϵ_i are zero-mean errors independent with the predictors. The regression function is assumed to have an additive form, and the effects of predictors are assumed to be linear. Wong et al. [29] proposed an estimation procedure for such models and applied it to a crop yield prediction application.

4.4.2 Selected Software Packages

fdapace is a counterpart of the PACE package, but it is written in R, free software which can be downloaded from <https://www.r-project.org/>. Although they are not exactly the same, the key functions are similar. Both FPCA and FPCReg functions we used for the analysis in this chapter can be used similarly in the *fdapace* package. Most of the methods used in this package are proposed for sparse and irregularly spaced functional data, but they may also be used for the analysis of dense or regularly spaced functional data. The core algorithm in this package allows users to implement the estimation of mean and covariance functions, eigenfunctions and PC scores, and prediction of continuous trajectories with point-wise confidence bands. The main smoothing method used in this package is the local polynomial smoothing.

fd is a package written in R. The functions in this package are aimed to perform the statistical analysis in FDA described in [1]. The classic book in FDA, [1], includes a variety of topics in FDA, such as FPCA, functional canonical correlation and discriminant analysis, functional linear models

with scalar or functional responses, and principal differential analysis. However, the book and its package mainly focus on the modeling and analysis of dense functional data rather than sparse functional data. *fd* mainly used splines and Fourier basis for smoothing and approximation.

refund, short for regression with functional data, is also a package written in R. It provides functions to perform different functional regressions, including function-on-scalar regression, scalar-on-function regression, and function-on-function regression. In particular, some functions in *refund* can be used to analyze image data. *refund* mainly uses splines and Fourier bases for smoothing and approximation, but it also provides wavelet-based functional regression methods with scalar responses and functional predictors. Multilevel FPCA is also available in *refund*.

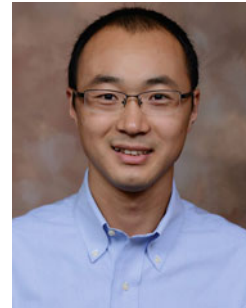
fd.usc, short for functional data analysis and utilities for statistical computing, is a R package. Except for traditional FPCA and functional regression, it also provides functions for supervised and non-supervised classification for functional data. It also provides an exploratory and descriptive analysis of functional data such as depth measurements and atypical curve detection. This package uses both spline smoothing and local polynomial smoothing.

FRegSigCom, short for functional regression using signal compression approach, is a package written in R. This package is mainly about various function-on-function regression methods, including linear function-on-function model with functional response and both scalar and functional predictors for a small or large number of functional, stepwise selection for function-on-function models with interactions, and some nonlinear function-on-function models. Particularly, their method can handle a large number of functional predictors and spiky functional data (see [30] for details). Splines and Fourier bases are mainly used for smoothing.

References

1. Ramsay, J.O., Silverman, B.W.: Functional Data Analysis. Springer, Berlin (2005)
2. Xu, Y., Qiu, Y., Schnable, J.C.: Functional Modeling of Plant Growth Dynamics. *The Plant Phenome Journal* **1**(1), 1–10 (2018)
3. Xu, Y., Li, Y., Nettleton, D.: Nested hierarchical functional data modeling and inference for the analysis of functional plant phenotypes. *J. Am. Stat. Assoc.* **113**(522), 593–606 (2018)
4. Yao, F., Müller, H.-G., Wang, J.-L.: Functional data analysis for sparse longitudinal data. *J. Am. Stat. Assoc.* **100**(470), 577–590 (2005)
5. Fan, J.: Local Polynomial Modelling and Its Applications: Monographs on Statistics and Applied Probability, vol. 66. Routledge, London (2018)
6. Rice, J.A., Silverman, B.W.: Estimating the mean and covariance structure nonparametrically when the data are curves. *J. R. Stat. Soc. Ser. B (Methodol.)* **53**(1), 233–243 (1991)
7. Yao, F., Lee, T.C.: Penalized spline models for functional principal component analysis. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **68**(1), 3–25 (2006)

8. Li, Y., Wang, N., Carroll, R.J.: Selecting the number of principal components in functional data. *J. Am. Stat. Assoc.* **108**(504), 1284–1294 (2013)
9. Yao, F., Müller, H.-G., Wang, J.-L.: Functional linear regression analysis for longitudinal data. *Ann. Stat.* **33**(6), 2873–2903 (2005)
10. Wang, J.-L., Chiou, J.-M., Müller, H.-G.: Functional data analysis. *Annu. Rev. Stat. Appl.* **3**, 257–295 (2016)
11. Hall, P., Müller, H.-G., Wang, J.-L.: Properties of principal component methods for functional and longitudinal data analysis. *Ann. Stat.* **34**(3), 1493–1517 (2006)
12. Hall, P., Vial, C.: Assessing the finite dimensionality of functional data. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **68**(4), 689–705 (2006)
13. Li, Y., Hsing, T.: Uniform convergence rates for nonparametric regression and principal component analysis in functional/longitudinal data. *Ann. Stat.* **38**(6), 3321–3351 (2010)
14. Li, Y., Hsing, T.: Deciding the dimension of effective dimension reduction space for functional and high-dimensional data. *Ann. Stat.* **38**(5), 3028–3062 (2010)
15. Zhang, X., Wang, J.-L.: From sparse to dense functional data and beyond. *Ann. Stat.* **44**(5), 2281–2321 (2016)
16. Liu, B., Müller, H.-G.: Estimating derivatives for samples of sparsely observed functions, with application to online auction dynamics. *J. Am. Stat. Assoc.* **104**(486), 704–717 (2009)
17. Dai, X., Müller, H.-G., Tao, W.: Derivative principal component analysis for representing the time dynamics of longitudinal and functional data. arXiv preprint arXiv:1707.04360 (2017)
18. Morris, J.S.: Functional regression. *Annual Review of Statistics and Its Application* **2**, 321–359 (2015)
19. Zhang, J.-T.: *Analysis of Variance for Functional Data*. Chapman and Hall/CRC, New York (2013)
20. Song, F.S., Montabon, F., Xu, Y.: The impact of national culture on corporate adoption of environmental management practices and their effectiveness. *Int. J. Prod. Econ.* **205**, 313–328 (2018)
21. Di, C.-Z., Crainiceanu, C.M., Caffo, B.S., Punjabi, N.M.: Multilevel functional principal component analysis. *Ann. Appl. Stat.* **3**(1), 458 (2009)
22. Sawant, P., Billor, N., Shin, H.: Functional outlier detection with robust functional principal component analysis. *Comput. Stat.* **27**(1), 83–102 (2012)
23. She, Y., Li, S., Wu, D.: Robust orthogonal complement principal component analysis. *J. Am. Stat. Assoc.* **111**(514), 763–771 (2016)
24. Cao, G., Yang, L., Todem, D.: Simultaneous inference for the mean function based on dense functional data. *Journal of Nonparametric Statistics* **24**(2), 359–377 (2012)
25. Cao, G., Wang, L., Li, Y., Yang, L.: Oracle-efficient confidence envelopes for covariance functions in dense functional data. *Stat. Sin.*, **26**, 359–383 (2016)
26. Müller, H.-G.: Functional modelling and classification of longitudinal data. *Scand. J. Stat.* **32**(2), 223–240 (2005)
27. James, G.M., Hastie, T.J.: Functional linear discriminant analysis for irregularly sampled curves. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **63**(3), 533–550 (2001)
28. Chiou, J.-M., Chen, Y.-T., Yang, Y.-F.: Multivariate functional principal component analysis: A normalization approach. *Stat. Sin.* **24**, 1571–1596 (2014)
29. Wong, R.K., Li, Y., Zhu, Z.: Partially linear functional additive models for multivariate functional data. *J. Am. Stat. Assoc.* **114**(525), 406–418 (2019)
30. Luo, R., Qi, X.: Function-on-function linear regression by signal compression. *J. Am. Stat. Assoc.* **112**(518), 690–705 (2017)



Yuhang Xu is an assistant professor in statistics in the Department of Applied Statistics and Operations Research at Bowling Green State University. He received his Ph.D. in Statistics from Iowa State University in 2016. His research interests include functional data analysis, survival analysis, measurement error, and interdisciplinary research in business, plant science, chemistry, etc.



Symmetric Geometric Skew Normal Regression Model

5

Debasis Kundu and Deepak Prajapati

Contents

5.1	Introduction	87
5.2	Geometric Skew Normal Distribution	89
5.3	Multivariate Geometric Skew Normal Distribution	94
5.4	Location Shift SGSN Distribution	95
5.4.1	Model Description	95
5.4.2	Maximum Likelihood Estimators	97
5.4.3	Testing of Hypothesis	98
5.5	SGSN Regression Model	98
5.6	Simulation Results	100
5.6.1	LS-SGSN Model	100
5.6.2	SGSN Regression Model	101
5.7	Real Data Analysis	101
5.7.1	LS-SGSN Model	101
5.7.2	SGSN Regression Model	102
5.8	Conclusions	102
	References	102

normal (MGSN) distribution also has several desirable properties. In this paper, we have proposed a symmetric geometric skew normal (SGSN) distribution as an alternative to a symmetric distribution like normal distribution, log Birnbaum-Saunders (BS) distribution, Student's t distribution, etc. It is a very flexible class of distributions, of which normal distribution is a special case. The proposed model has three unknown parameters, and it is observed that the maximum likelihood (ML) estimators of the unknown parameters cannot be obtained in explicit forms. In this paper, we have proposed a very efficient expectation maximization (EM) algorithm, and it is observed that the proposed EM algorithm works very well. We have further considered a location shift SGSN regression model. It is a more flexible than the standard Gaussian regression model. The ML estimators of the unknown parameters are obtained based on EM algorithm. Extensive simulation experiments and the analyses of two data sets have been presented to show the effectiveness of the proposed model and the estimation techniques.

Abstract

Recently, Kundu (2014, Sankhya, Ser. B, 167–189, 2014) proposed a geometric skew normal (GSN) distribution as an alternative to Azzalini's skew normal (ASN) distribution. The GSN distribution can be a skewed distribution; it can be heavy tailed as well as multimodal also, unlike ASN distribution. It can be easily extended to the multivariate case also. The multivariate geometric skew

Keywords

Absolute continuous distribution · Singular distribution · Fisher information matrix · EM algorithm · Joint probability distribution function · Joint probability density function

D. Kundu (✉)
Department of Mathematics and Statistics, Indian Institute of Technology Kanpur, Kanpur, Uttar Pradesh, India
e-mail: kundu@iitk.ac.in

D. Prajapati
Decision Sciences Area, Indian Institute of Management, Lucknow, India
e-mail: deepak.prajapati@iiml.ac.in

5.1 Introduction

Azzalini [3] introduced a skew normal distribution which has received considerable attention in the last three decades. It has three parameters, and it is a skewed distribution of which normal distribution is a special case. From now on we call it as the Azzalini's skew normal (ASN) distribution. A three-parameter ASN distribution has the following probability density function (PDF):

$$f(x) = 2\phi\left(\frac{x-\mu}{\sigma}\right)\Phi\left(\frac{\lambda(x-\mu)}{\sigma}\right).$$

Here $\phi(\cdot)$ is the PDF of a standard normal distribution, the $\Phi(\cdot)$ is the corresponding cumulative distribution function (CDF), $-\infty < \mu < \infty$ is the location parameter, $\sigma > 0$ is the scale parameter, and $-\infty < \lambda < \infty$ is the skewness or tilt parameter. Note that when $\lambda = 0$, it becomes a normal PDF with mean μ and standard deviation σ . It has some interesting physical interpretation also as a hidden truncation model; see, for example, Arnold et al. [2] and Arnold and Beaver [1]. It is a very flexible class of distribution functions, and due to which it has been used to analyze various skewed data sets. Although it has several desirable properties, it has been observed that the maximum likelihood (ML) estimators of the unknown parameters may not always exist; see, for example, Gupta and Gupta [10]. It can be shown that for any sample size if the data come from an ASN distribution, there is a positive probability that the MLEs do not exist. The problem becomes more severe for higher dimension. Moreover, the PDF of an ASN distribution is always unimodal and thin tailed. Due to these limitations, ASN distribution cannot be used for analyzing for a large class of skewed data sets.

To overcome the problem of the ASN distribution, Gupta and Gupta [10] proposed the power normal distribution which has the following CDF:

$$F(x) = \left[\Phi\left(\frac{x-\mu}{\sigma}\right) \right]^\alpha.$$

Here also, $-\infty < \mu < \infty$ is the location parameter, $\sigma > 0$ is the scale parameter, and $\alpha > 0$ is the skewness parameter. When $\alpha = 1$, it becomes a normal distribution function with mean μ and standard deviation σ . Therefore, in this case also the normal distribution can be obtained as a special case. It has been shown by Gupta and Gupta [10] that for $\alpha > 1$ it is positively skewed and, for $\alpha < 1$, it is negatively skewed. They have obtained several other interesting properties of the power normal distribution in the same paper. Kundu and Gupta [18] provided an efficient estimation procedure and also defined bivariate power normal distribution. Although the power normal distribution as proposed by Gupta and Gupta [10] has several desirable properties, and the ML estimators also always exist, it is always unimodal and thin tailed similar to the ASN distribution. Therefore, if the data indicate that the observations are coming from a heavy tailed or multimodal distribution, it can be used to analyze that data set.

Recently, Kundu [14] introduced a three-parameter geometric skew normal (GSN) distribution as an alternative to the popular Azzalini's [3] skew normal (ASN) distribution or the power normal distribution of Gupta and Gupta [10]. The GSN distribution can be defined as follows. Suppose X_1, X_2, \dots , are independent and identically distributed (i.i.d.)

normal random variables with mean μ and variance σ^2 , and N is a geometric random variable with parameter p . Here, a geometric random variable with parameter p will be denoted by $GE(p)$, and it has the following probability mass function:

$$P(N = n) = p(1-p)^{n-1}; \quad n = 1, 2, \dots \quad (5.1)$$

It is assumed that N and X_i 's are independently distributed. The random variable

$$X \stackrel{d}{=} \sum_{i=1}^N X_i,$$

is said to have a GSN distribution with the parameters μ , σ , and p . From now on it will be denoted by $GSN(\mu, \sigma, p)$.

The GSN distribution also has three parameters similar to the ASN or power normal distribution. But the main advantage of the GSN distribution over the ASN or the power normal distribution is that the GSN distribution is more flexible than them in the sense its PDF can have more variety of shapes compared to the ASN distribution. The PDF of the GSN distribution can be symmetric, skewed, unimodal, bimodal, and multimodal shaped also. Moreover, the GSN distribution can be heavy tailed also depending on the parameter values. In case of an ASN distribution, it is observed that the ML estimators may not always exist. In fact it can be shown that the ML estimates will not exist if the all the data points are of the same sign. But in case of the GSN distribution, the ML estimates of the unknown parameters exist, if the sample size is greater than three. If $\mu = 0$, it becomes a symmetric distribution, and it is called a symmetric GSN (SGSN) distribution. From now on a $GSN(\mu, \sigma, p)$ with $\mu = 0$ will be denoted by $SGSN(\sigma, p)$.

As GSN has been introduced, along the same line, multivariate geometric skew normal (MGSN) distribution has been introduced by Kundu [17] as an alternative to Azzalini's multivariate skew normal distribution. In this case, the marginals are GSN distributions, and the joint PDF can be unimodal, bimodal, and multimodal also. Multivariate normal distribution can be obtained as a special case. It has also several interesting properties, and the moments and product moments can be obtained quite conveniently from the joint characteristic function, which can be expressed in explicit form. It has several characterization properties similar to the multivariate normal distribution. A brief review of the MGSN distribution will be provided in Sect. 5.3.

In recent times, the Birnbaum-Saunders (BS) distribution has received a considerable amount of attention in the statistical and some related literature. The BS distribution was originally derived by Birnbaum and Saunders [7] by showing that the fatigue failure is caused by the development and growth of cracks from the cyclic loading. The BS distribution can be defined as follows. Suppose T is a nonnegative random

variable and T is said to have a BS distribution with shape parameter α and scale parameter β ; it will be denoted by $BS(\alpha, \beta)$, if the cumulative distribution function (CDF) of T is given by

$$F_{BS}(t; \alpha, \beta) = \Phi \left\{ \frac{1}{\alpha} \left(\sqrt{\frac{t}{\beta}} - \sqrt{\frac{\beta}{t}} \right) \right\}, \quad t > 0, \quad (5.2)$$

and zero, otherwise. Here $\Phi(\cdot)$ is the CDF of a standard normal distribution. If T follows (\sim) , $BS(\alpha, \beta)$, then $Y = \ln T$ is said to have a log-BS (LBS) distribution. The CDF of Y can be written as

$$F_{LBS}(y; \alpha, \beta) = \Phi \left\{ \frac{2}{\alpha} \sinh \left(\frac{y - \ln \beta}{2} \right) \right\}, \quad -\infty < y < \infty, \quad (5.3)$$

and it will be denoted by $LBS(\alpha, \beta)$. The LBS distribution was originally proposed by Rieck [25]; see also Rieck and Nedelman [26] and Kundu [15, 16] in this respect. It has been observed by Rieck [25] that the LBS distribution is symmetric, it is strongly unimodal for $\alpha < 2$ and for $\alpha \geq 2$, and it is bimodal. It may be mentioned that SGSN distribution is more flexible than the LBS distribution in the sense. SGSN distribution can be heavy tailed also, whereas LBS distribution can never be a heavy tailed. For a comprehensive review on BS distribution, one is referred to the review article by Balakrishnan and Kundu [6]. Although LBS distribution can be bimodal and the ML estimators of the unknown parameters always exist, it cannot be multimodal or heavy tailed.

In this paper, we have introduced a location-shift SGSN (LS-SGSN) distribution. It is observed that due to the presence of three parameters, it is more flexible than a two-parameter normal distribution. Moreover, the normal distribution can be obtained as a special case of the LS-SGSN distribution. We have provided several properties of the LS-SGSN distribution and obtained the characteristic function and different moments. It is interesting to observe that although the PDF of a SGSN distribution can be obtained as an infinite series, the characteristic function can be obtained in explicit form. The LS-SGSN distribution has three parameters. The ML estimators cannot be obtained in explicit forms; they have to be obtained by solving a three-dimensional optimization problem. Some standard numerical methods like Newton-Raphson or Gauss-Newton may be used to compute the ML estimates. But it involves providing efficient initial guesses; otherwise, the algorithm may not converge, and even if it converges it may converge to a local minimum rather than a global minimum. To avoid that, we have treated this problem as a missing value problem, and we have used the EM algorithm to compute the ML estimators. It is observed that at each ‘‘E’’-step, the corresponding ‘‘M’’-step can be performed explicitly. Hence,

no optimization problem needs to be solved numerically at the ‘‘M’’-step, and they are unique. Moreover, at the last step of the EM algorithm, one can easily obtain the observed Fisher information matrix based on the method of Louis [22]. Hence, the confidence intervals of the unknown parameters also can be obtained quite conveniently. We have performed extensive simulation experiments, and the analysis of one data set has been presented for illustrative purposes.

We have further considered the multiple linear regression model in the presence of additive SGSN errors. Note that the analysis of multiple linear regression model in the presence of additive ASN distribution has been well studied in the statistical literature both from the classical and Bayesian viewpoints; see, for example, Sahu et al. [27], Lachos et al. [20], and the references cited therein. Similarly, log-linear BS regression model also has been well studied in the literature; see, for example, Zhang et al. [30] and Balakrishnan and Kundu [6]. It is expected that the proposed multiple regression model will be more flexible than the ASN and LBS multiple regression models.

In this paper, we provide the likelihood inference of the unknown parameters of the proposed SGSN regression model. It is observed that the ML estimators of the unknown parameters cannot be obtained in closed forms. In this case, also we have used a very efficient EM algorithm to avoid solving nonlinear optimization problem. The implementation of the proposed EM algorithm is quite simple in practice. Moreover, using the method of Louis [22] at the last step of the EM algorithm, the observed Fisher information matrix also can be obtained in a standard manner. Hence, the confidence intervals of the unknown parameters based on the observed Fisher information matrix also can be constructed. Simulation experimental results and analysis of a data set have been presented.

The rest of the paper is organized as follows. In Sects. 5.2 and 5.3, we provide a brief review of the GSN distribution and MGSN distribution, respectively. The SGSN and LS-SGSN distributions have been introduced, and the EM algorithm has been discussed in Sect. 5.4. In Sect. 5.5, we have formulated the LS-SGSN regression model and discussed the corresponding EM algorithm. Simulation results have been presented in Sect. 5.6, and the analysis of two data sets has been presented in Sect. 5.7. Finally, we conclude the paper in Sect. 5.8.

5.2 Geometric Skew Normal Distribution

In the previous section, we have provided the definition of a GSN distribution. In this section, we provide a brief review and some properties of a GSN distribution. All the details can be obtained in Kundu [14]. If $X \sim GSN(\mu, \sigma, p)$, then the CDF and PDF of X become

$$F_{GSN}(x; \mu, \sigma, p) = p \sum_{k=1}^{\infty} \Phi\left(\frac{x - k\mu}{\sigma\sqrt{k}}\right) (1-p)^{k-1}, \quad (5.4)$$

and

$$f_{GSN}(x; \mu, \sigma, p) = \sum_{k=1}^{\infty} \frac{p}{\sigma\sqrt{k}} \phi\left(\frac{x - k\mu}{\sigma\sqrt{k}}\right) (1-p)^{k-1}, \quad (5.5)$$

respectively, for $x \in \mathbb{R}$. Here $\phi(\cdot)$ denotes the PDF of a standard normal distribution function.

The PDF of the GSN distribution (5.5) can take a variety of shapes. It can be symmetric, positively skewed, negatively skewed, unimodal, bimodal, multimodal, and heavy tailed also. For different shapes of the PDF of a GSN distribution, see Fig. 5.1 and also Kundu [14]. It is clear that it can be positively skewed, negatively skewed, unimodal, multimodal, and heavy tailed also depending on the parameter values.

If $X \sim \text{GSN}(\mu, \sigma, p)$, then the characteristic function of X is

$$\phi_X(t) = E(e^{itX}) = \frac{pe^{(i\mu t - \frac{\sigma^2 t^2}{2})}}{1 - (1-p)e^{(i\mu t - \frac{\sigma^2 t^2}{2})}}, \quad t \in \mathbb{R}.$$

The mean and variance of X are

$$E(X) = \frac{\mu}{p} \quad \text{and} \quad V(X) = \frac{\mu^2(1-p) + \sigma^2 p}{p^2}.$$

Any higher order moment can be obtained as infinite series as follows:

$$E(X^m) = p \sum_{n=1}^{\infty} (1-p)^{n-1} c_m(n\mu, n\sigma^2).$$

Here, $c_m(n\mu, n\sigma^2) = E(Y^m)$, where $Y \sim N(n\mu, n\sigma^2)$. It may be mentioned that c_m can be obtained using confluent hypergeometric function; see, for example, Johnson, Kotz, and Balakrishnan [13]. The skewness of a GSN(μ, σ, p) can be written as

$$\gamma_1 = \frac{(1-p)(\mu^3(2p^2 - p + 1) + 2\mu^2 p^2 + \mu\sigma^2(3-p)p)}{(\sigma^2 p + \mu^2(1-p))^{3/2}}.$$

Hence, unlike normal distribution, the skewness depends on all the three parameters.

If $\mu = 0$, it becomes symmetric, and the CDF and PDF of a SGSN distribution become

$$F_{SGSN}(x; \sigma, p) = p \sum_{k=1}^{\infty} \Phi\left(\frac{x}{\sigma\sqrt{k}}\right) (1-p)^{k-1}, \quad (5.6)$$

and

$$f_{SGSN}(x; \sigma, p) = \sum_{k=1}^{\infty} \frac{p}{\sigma\sqrt{k}} \phi\left(\frac{x}{\sigma\sqrt{k}}\right) (1-p)^{k-1}, \quad (5.7)$$

respectively, for $x \in \mathbb{R}$. The PDF of SGSN are provided in Fig. 5.2 for different σ and p values.

It is clear from Fig. 5.2 that the tail probabilities of a SGSN distribution increase as p decreases. It behaves like a heavy tailed distribution as p tends to zero. As p tends to one, it behaves like a normal distribution.

If $X \sim \text{SGSN}(\sigma, p)$, then the characteristic function of X can be obtained as

$$\phi_X(t) = E(e^{itX}) = \frac{pe^{(-\frac{\sigma^2 t^2}{2})}}{1 - (1-p)e^{(-\frac{\sigma^2 t^2}{2})}}, \quad t \in \mathbb{R}.$$

Using the characteristic function or otherwise, the mean and variance of X can be expressed as follows:

$$E(X) = 0 \quad \text{and} \quad V(X) = \frac{\sigma^2}{p}.$$

For higher moments, it can be easily observed from the characteristic function that if $X \sim \text{SGSN}(1, p)$, then

$$E(X^m) = pd_m \sum_{n=1}^{\infty} (1-p)^{n-1} n^{m/2},$$

where $d_m = E(Z^m)$, $Z \sim N(0, 1)$, and

$$d_m = \begin{cases} 0 & \text{if } m \text{ is odd} \\ \frac{2^{m/2} \Gamma(\frac{m+1}{2})}{\sqrt{\pi}} & \text{if } m \text{ is even.} \end{cases}$$

In Fig. 5.3, we have plotted the characteristic functions of SGSN for different parameter values. Just to show how different it can be from a normal distribution, we have plotted the characteristic function of the corresponding normal distribution with mean zero and variance σ^2/p . It is clear that the characteristic functions can be quite different particularly for small values of p . But as p approaches one, they become close to each other.

It can be easily seen that the variance diverges as p approaches zero. The GSN distribution is known to be infinitely divisible and geometrically stable. For different other properties and estimation methods, interested readers are referred to the original article of Kundu [14].

The following derivations are needed for further development. These results will be used to develop EM algorithm for SGSN distribution. Suppose $X \sim \text{SGSN}(\sigma, p)$ and N is the associated GE(p) random variable, then for $0 < p < 1$, and $-\infty < x < \infty$, $n = 1, 2, \dots$,

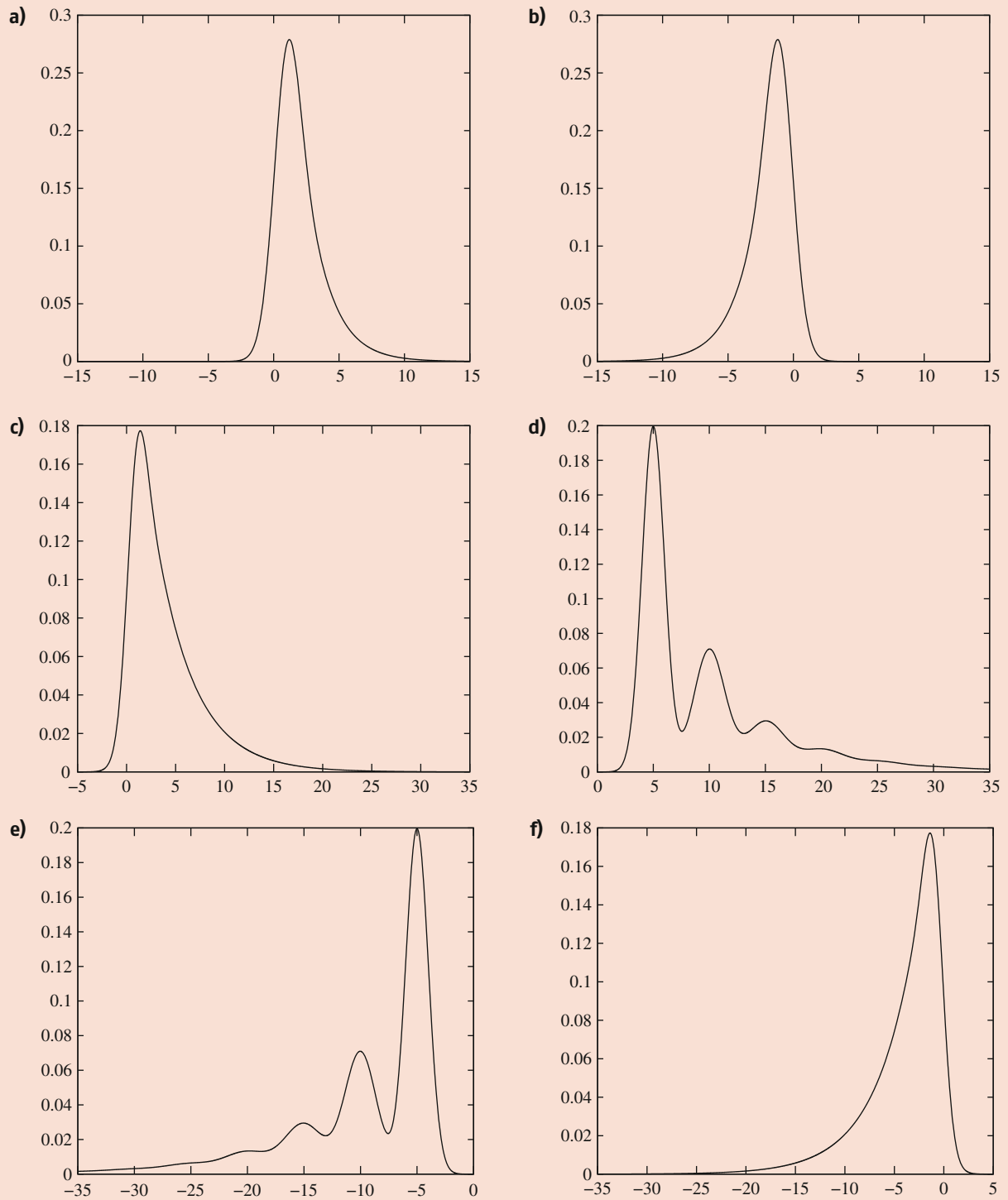


Fig. 5.1 PDF plots of $\text{GSN}(\mu, \sigma, p)$ distribution for different (μ, σ, p) values: (a) $(1.0, 1.0, 0.5)$ (b) $(-1.0, 1.0, 0.5)$ (c) $(1.0, 1.0, 0.25)$ (d) $(5.0, 1.0, 0.5)$ (e) $(-5.0, 1.0, 0.5)$ (f) $(-1.0, 1.0, 0.25)$

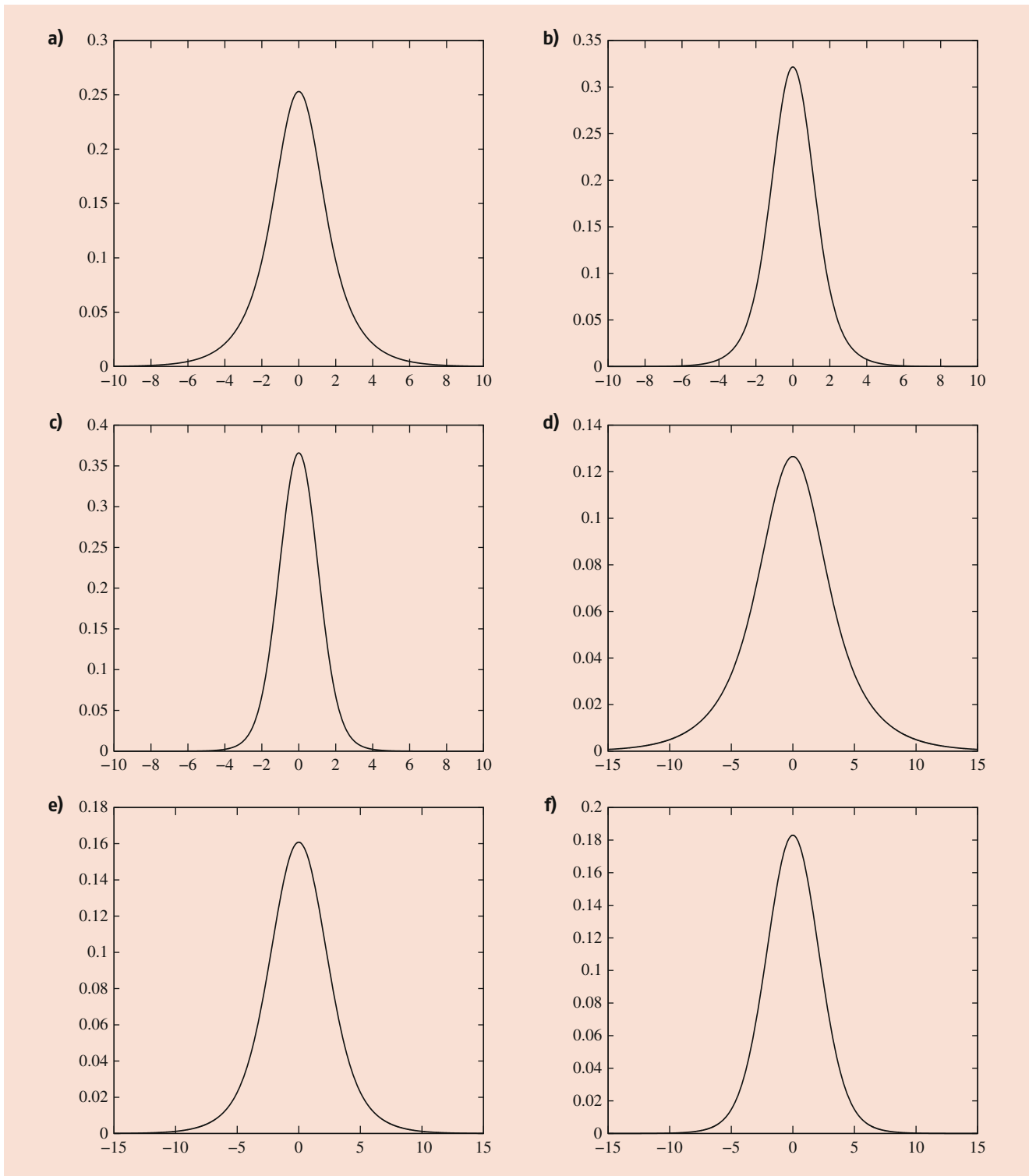


Fig. 5.2 PDF plots of $SGSN(\sigma, p)$ distribution for different (σ, p) values: (a) (1.0,0.25) (b) (1.0,0.50) (c) (1.0,0.75) (d) (2.0,0.25) (e) (2.0,0.50) (f) (2.0,0.75)

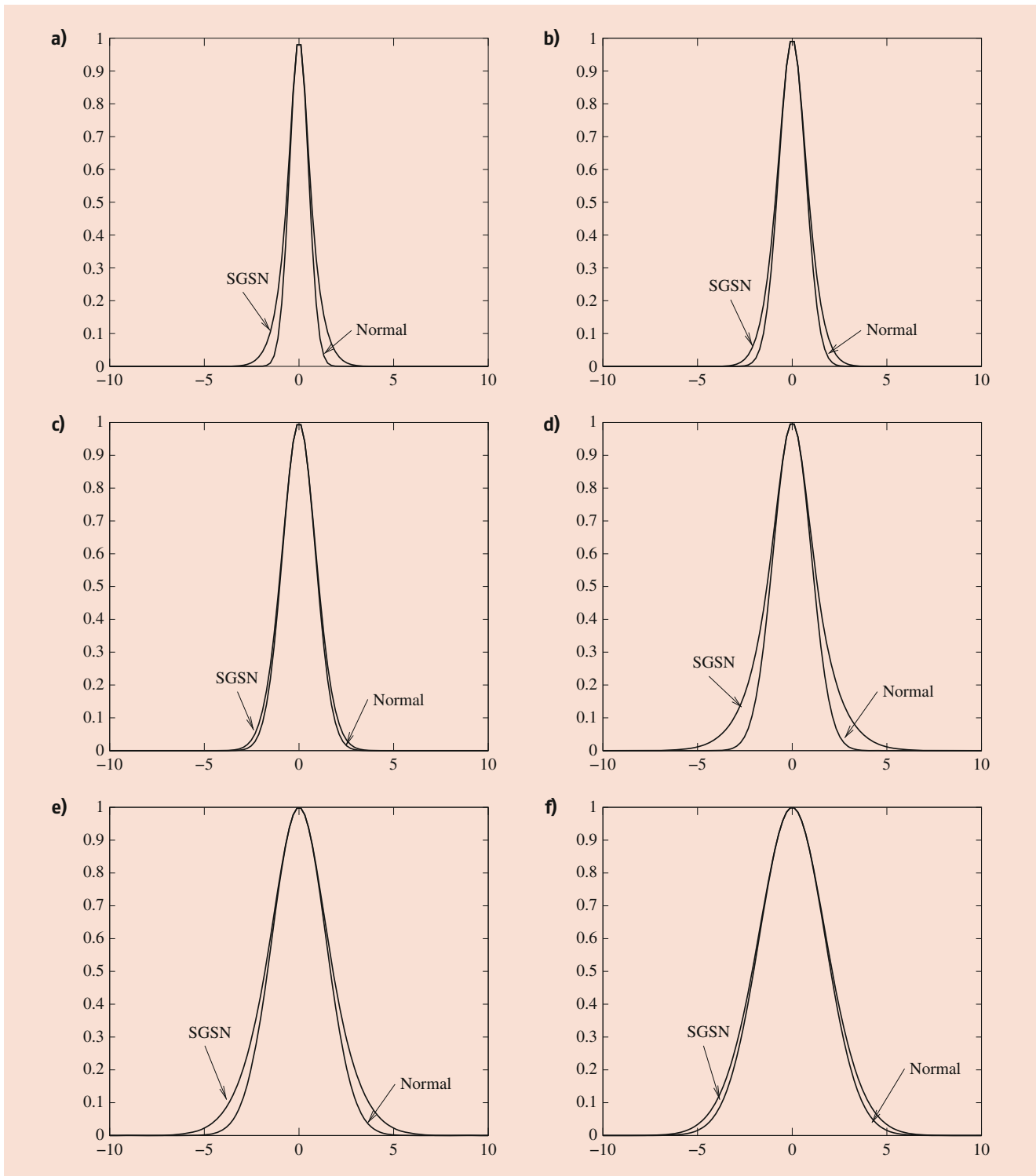


Fig. 5.3 CHF plots of $\text{SGSN}(\sigma, p)$ distribution for different (σ, p) values: (a) (1.0,0.25) (b) (1.0,0.50) (c) (1.0,0.75) (d) (0.5,0.25) (e) (0.5,0.50) (f) (0.5,0.75)

$$\begin{aligned} P(X \leq x, N = n) &= P(X \leq x|N = n)P(N = n) \\ &= p(1-p)^{n-1}\Phi\left(\frac{x}{\sigma\sqrt{n}}\right), \end{aligned} \quad (5.8)$$

for $p = 1$,

$$P(X \leq x, N = n) = \begin{cases} \Phi\left(\frac{x}{\sigma}\right) & \text{if } n = 1 \\ 0 & \text{if } n > 1. \end{cases} \quad (5.9)$$

Hence, the joint PDF, $f_{X,N}(x, n)$, of (X, N) for $0 < p < 1$, $-\infty < x < \infty$, $n = 1, 2, \dots$, becomes

$$f_{X,N}(x, n) = p(1-p)^{n-1} \frac{1}{\sigma\sqrt{2\pi n}} e^{-\frac{x^2}{2n\sigma^2}},$$

and for $p = 1$,

$$f_{X,N}(x, n) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} & \text{if } n = 1 \\ 0 & \text{if } n > 1. \end{cases} \quad (5.10)$$

Therefore,

$$P(N = n|X = x) = \frac{(1-p)^{n-1} e^{-\frac{x^2}{2n\sigma^2}} / \sqrt{n}}{\sum_{j=1}^{\infty} (1-p)^{j-1} e^{-\frac{x^2}{2j\sigma^2}} / \sqrt{j}}. \quad (5.11)$$

It can be easily shown that (5.11) is a unimodal function in n . Hence, there exists a unique n_0 such that $P(N = n_0|X = x) > P(N = n|X = x)$, for any $n \neq n_0$. In fact, n_0 can be obtained as the minimum value of $n \geq 1$, such that

$$\frac{P(N = n+1|X = x)}{P(N = n|X = x)} = \frac{\sqrt{n}(1-p)}{\sqrt{n+1}} e^{\frac{x^2}{2\sigma^2} \left(\frac{1}{n} - \frac{1}{n+1} \right)} < 1. \quad (5.12)$$

Moreover, from (5.11) one can easily obtain

$$E(N|X = x) = \frac{\sum_{n=1}^{\infty} \sqrt{n}(1-p)^{n-1} e^{-\frac{x^2}{2n\sigma^2}}}{\sum_{j=1}^{\infty} (1-p)^{j-1} e^{-\frac{x^2}{2j\sigma^2}} / \sqrt{j}}, \quad (5.13)$$

and

$$E(1/N|X = x) = \frac{\sum_{n=1}^{\infty} n^{-3/2}(1-p)^{n-1} e^{-\frac{x^2}{2n\sigma^2}}}{\sum_{j=1}^{\infty} (1-p)^{j-1} e^{-\frac{x^2}{2j\sigma^2}} / \sqrt{j}}. \quad (5.14)$$

5.3 Multivariate Geometric Skew Normal Distribution

Azzalini and Dalla Valle [5] introduced a multivariate distribution with ASN marginals. We call it as multivariate ASN (MASN) distribution, and it can be defined as follows:

A random vector $\mathbf{Z} = (Z_1, \dots, Z_d)^\top$ is a d-variate MASN distribution, if it has the following PDF:

$$g(\mathbf{z}) = 2\phi_d(\mathbf{z}, \mathbf{\Omega})\Phi(\boldsymbol{\alpha}^\top \mathbf{z}), \quad \mathbf{z} \in \mathbb{R}^d,$$

where $\phi_d(\mathbf{z}, \mathbf{\Omega})$ denotes the PDF of a d-variate multivariate normal distribution with standardized marginals and correlation matrix $\mathbf{\Omega}$. Here the parameter vector $\boldsymbol{\alpha}$ is known as the shape vector, and depending on the shape vector, the PDF of a MASN distribution can take a variety of shapes. The PDF is always unimodal, and when $\boldsymbol{\alpha} = \mathbf{0}$, then \mathbf{Z} has the multivariate normal distribution with mean vector $\mathbf{0}$ and correlation matrix $\mathbf{\Omega}$. Although MASN is a very flexible distribution, but if the marginals are heavy tailed or multimodal, it cannot be used; see, for example, the excellent recent monograph by Azzalini and Capitanio [4]. Moreover, if the data come from a MASN distribution, it can be shown that the MLEs do not exist with a positive probability. Due to these reasons, several kernels instead of normal kernel have been used, but they have their own problems.

Kundu [17] proposed MGSN distribution along the same line as the GSN distribution. The MGSN distribution has been defined as follows. Let us use the following notations. A d-variate normal random variable with mean vector $\boldsymbol{\mu}$ and the dispersion matrix $\boldsymbol{\Sigma}$ will be denoted by $N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The corresponding PDF and CDF will be denoted $\phi_d(x; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $\Phi_d(x; \boldsymbol{\mu}, \boldsymbol{\Sigma})$, respectively. Now a d-variate MGSN distribution can be defined as the following. Suppose X_i for $i = 1, 2, 3, \dots$ are i.i.d. $N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $N \sim \text{GE}(p)$ and they are independently distributed. Then

$$X = \sum_{i=1}^N X_i$$

is said to have MGSN distribution with parameters $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, and p and will be denoted by $\text{MGSN}_d(\boldsymbol{\mu}, \boldsymbol{\Sigma}, p)$.

MGSN is a very flexible multivariate distribution, and its joint PDF can take a variety of shapes. It can be unimodal, bimodal, and multimodal also. Since the marginals are GSN distributions, therefore the marginals can be positively and negatively skewed, and it can be heavy tailed also. If $\mathbf{X} \sim \text{MGSN}_d(\boldsymbol{\mu}, \boldsymbol{\Sigma}, p)$, then the PDF and CDF of X can be written as

$$f_X(x, \boldsymbol{\mu}, \boldsymbol{\Sigma}, p) = \sum_{k=1}^{\infty} \frac{p(1-p)^{k-1}}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2} k^{d/2}} e^{-\frac{1}{2k} (x-k\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (x-k\boldsymbol{\mu})}$$

and

$$F_X(x, \boldsymbol{\mu}, \boldsymbol{\Sigma}, p) = \sum_{k=1}^{\infty} p(1-p)^{k-1} \Phi_d(x; k\boldsymbol{\mu}, k\boldsymbol{\Sigma}),$$

respectively. When $\boldsymbol{\mu} = \mathbf{0}$, the PDF and CDF of \mathbf{X} can be written as

$$f_{\mathbf{X}}(\mathbf{x}, \mathbf{0}, \boldsymbol{\Sigma}, p) = \sum_{k=1}^{\infty} \frac{p(1-p)^{k-1}}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2} k^{d/2}} e^{-\frac{1}{2k} \mathbf{x}^{\top} \boldsymbol{\Sigma}^{-1} \mathbf{x}}$$

and

$$F_{\mathbf{X}}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, p) = \sum_{k=1}^{\infty} p(1-p)^{k-1} \Phi_d(\mathbf{x}; \mathbf{0}, k\boldsymbol{\Sigma}),$$

respectively.

Similar to the GSN distribution, when $p = 1$, MGSN distribution becomes multivariate normal distribution. The PDF of a MGSN distribution can take a variety of shapes. Contour plots of the PDF of a MGSN distribution for different parameter values when $d = 2$ are provided in Fig. 5.4. It is clear that it can be positively and negative skewed, unimodal, bimodal, multimodal, heavy tailed depending on the parameter values. When $\boldsymbol{\mu} = \mathbf{0}$, then it becomes a symmetric distribution.

If $\mathbf{X} \sim \text{MGSN}_d(\boldsymbol{\mu}, \boldsymbol{\Sigma}, p)$, then the characteristic function of \mathbf{X} is

$$\phi_{\mathbf{X}}(\mathbf{t}) = \frac{p e^{i\boldsymbol{\mu}^{\top} \mathbf{t} + \frac{1}{2} \mathbf{t}^{\top} \boldsymbol{\Sigma} \mathbf{t}}}{1 - (1-p) e^{i\boldsymbol{\mu}^{\top} \mathbf{t} + \frac{1}{2} \mathbf{t}^{\top} \boldsymbol{\Sigma} \mathbf{t}}}, \quad \mathbf{t} \in \mathbb{R}^d.$$

Since the characteristic function is in a compact form, many properties can be derived quite conveniently. Different moments, product moments, and multivariate skewness can be obtained using the characteristic function; see Kundu [17] for details.

MGSN distribution has several properties like multivariate normal distribution. For example, if \mathbf{X} is a d -variate MGSN distribution, then if we partition \mathbf{X} as

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{pmatrix},$$

where \mathbf{X}_1 and \mathbf{X}_2 are of the orders d_1 and $n - d_1$, respectively, then \mathbf{X}_1 is d_1 -variate MGSN distribution and \mathbf{X}_2 is a $n - d_1$ variate MGSN distribution. Similarly, if \mathbf{X} is d -variate MGSN distribution and \mathbf{D} is a $s \times d$ matrix, with rank $s \leq d$, then $\mathbf{Z} = \mathbf{D}\mathbf{X}$ is a s -variate MGSN distribution. It has been shown that \mathbf{X} is d -variate MGSN distribution if and only if $Y = \mathbf{c}^{\top} \mathbf{X}$ is a GSN distribution, for all $\mathbf{c} \in \mathbb{R}^d \neq \mathbf{0}$. Several other interesting properties including canonical correlation, majorization, and characterization have been provided in Kundu [17]. Estimation of the unknown parameters is an important problem. A d -variate MGSN distribution has $1 + d + d(d + 1)/2$ unknown parameters. The usual ML computation involves solving a $1 + d + d(d + 1)/2$ variate

optimization problem. Therefore, even for $d = 3$, it involves solving a ten-dimensional optimization problem, and clearly it is a nontrivial problem. A very efficient EM algorithm has been proposed by Kundu [17], which does not require solving any optimization problem, i.e., at each ‘‘E’’-step, the corresponding ‘‘M’’-step can be performed explicitly. Due to this reason, MGSN can be used quite efficiently in practice even for large d .

5.4 Location Shift SGSN Distribution

5.4.1 Model Description

Now we will introduce location shift SGSN distribution, which can be used quite effectively for analyzing symmetric data as an alternative to any symmetric distribution such as normal, log-BS, Student’s t distributions, etc. A random variable Y is called a location shift SGSN (LS-SGSN) distribution if

$$Y = \theta + X, \quad (5.15)$$

where $\theta \in \mathbb{R}$ and $X \sim \text{SGSN}(\sigma, p)$. If the random variable Y has the form (5.15), then it will be denoted by LS-SGSN(θ, σ, p). If $Y \sim \text{LS-SGSN}(\theta, \sigma, p)$, then the CDF and PDF of Y become

$$F_Y(y; \theta, \sigma, p) = p \sum_{k=1}^{\infty} \Phi\left(\frac{y - \theta}{\sigma\sqrt{k}}\right) (1-p)^{k-1}, \quad (5.16)$$

and

$$f_Y(y; \theta, \sigma, p) = \sum_{k=1}^{\infty} \frac{p}{\sigma\sqrt{k}} \phi\left(\frac{y - \theta}{\sigma\sqrt{k}}\right) (1-p)^{k-1}, \quad (5.17)$$

respectively. Clearly, LS-SGSN distribution will be a more flexible than a normal distribution due to the presence of an extra parameter. Moreover, normal distribution can be obtained as a special case of the LS-SGSN distribution. We need the following derivations for further development, when $0 < p < 1$, $y \in \mathbb{R}$, and $n = 1, 2, \dots$:

$$\begin{aligned} P(Y \leq y, N = n) &= P(X \leq y - \theta, N = n) \\ &= p(1-p)^{n-1} \Phi\left(\frac{y - \theta}{\sigma\sqrt{n}}\right). \end{aligned}$$

and

$$f_{Y,N}(y, n) = p(1-p)^{n-1} \frac{1}{\sigma\sqrt{2\pi n}} e^{-\frac{(y-\theta)^2}{2\sigma^2 n}}.$$

If $Y \sim \text{LS-SGSN}(\theta, \sigma, p)$, then the characteristic function of Y can be obtained as

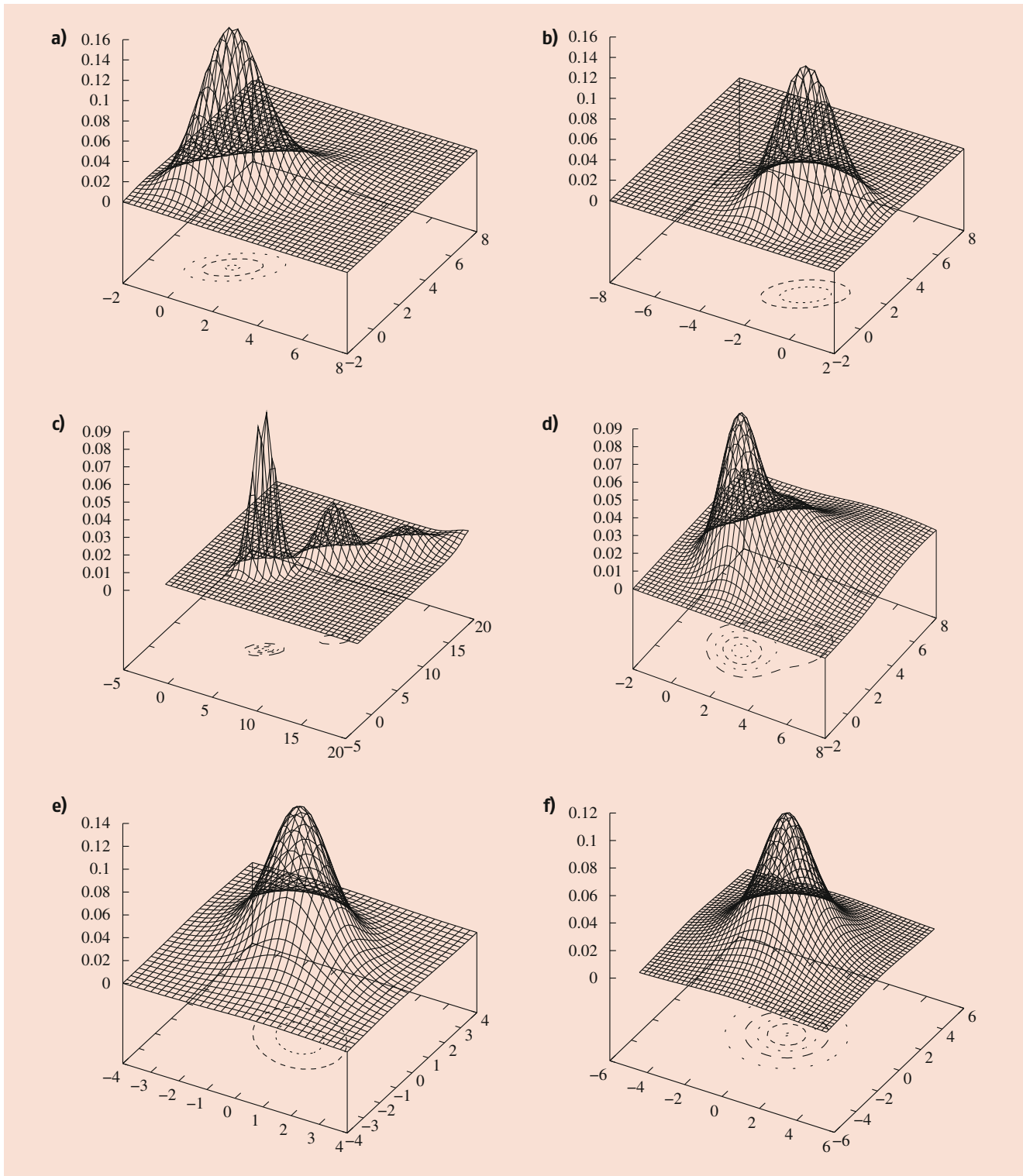


Fig. 5.4 PDF plots of $MGSN_2(\mu, \Sigma, p)$ distribution for different $(\mu_1, \mu_2, \sigma_{11}, \sigma_{22}, \sigma_{12}, p)$ values: (a) (1.0,1.0,1.0,0.5,0.75) (b) (-1.0,1.0,1.0,1.0,0.5,0.75) (c) (5,5,1.0,1.0,0.5,0.5) (d) (1.5,1.5,1.0,1.0,-0.25,0.5) (e) (0.0,0.0,1.0,1.0,-0.25,0.5) (f) (0.0, 0.0, 1.0, 1.0, 0.0, 0.25)

$$\phi_Y(t) = E(e^{itY}) = e^{it\theta} \frac{pe^{-\frac{\sigma^2 t^2}{2}}}{1 - (1-p)e^{-\frac{\sigma^2 t^2}{2}}}, \quad t \in \mathbb{R}.$$

The mean and variance of Y are

$$E(Y) = \theta \quad \text{and} \quad V(Y) = \frac{\sigma^2}{p},$$

respectively.

5.4.2 Maximum Likelihood Estimators

The proposed LS-SGSN distribution has three parameters. Now we will discuss about the estimation of the unknown parameters of the LS-SGSN distribution and the associated confidence intervals.

Suppose we have a random sample of size n , say,

$$\mathcal{D}_1 = \{x_1, \dots, x_n\}$$

from LS-SGSN(θ, σ, p). The log-likelihood function based on the observation \mathcal{D}_1 can be written as

$$l(\theta, \sigma, p) = n \ln p - n \ln \sigma + \sum_{i=1}^n \ln \left\{ \sum_{k=1}^{\infty} \frac{1}{\sqrt{k}} \phi \left(\frac{x_i - \theta}{\sigma \sqrt{k}} \right) (1-p)^{k-1} \right\}. \quad (5.18)$$

Therefore, the ML estimators of the unknown parameters can be obtained by maximizing (5.18) with respect to the unknown parameters. The normal equations can be obtained as

$$\begin{aligned} \dot{l}_\theta &= \frac{\partial l(\theta, \sigma, p)}{\partial \theta} = 0, \quad \dot{l}_\sigma = \frac{\partial l(\theta, \sigma, p)}{\partial \sigma} = 0, \\ \dot{l}_p &= \frac{\partial l(\theta, \sigma, p)}{\partial p} = 0. \end{aligned} \quad (5.19)$$

Hence, the ML estimates can be obtained by solving all the three normal equations in (5.19) simultaneously. Clearly, the ML estimates cannot be obtained in explicit forms, and one needs to use some iterative methods to solve (5.19). One may use the Newton-Raphson-type algorithm to solve these nonlinear equations, but it has its own problem of local convergence and the choice of initial values. To avoid that, we have proposed to use EM-type algorithm to compute the ML estimates. It is observed that at each E-step the corresponding M-step can be obtained explicitly. Hence, it can be implemented very easily.

The basic idea of the proposed EM-type algorithm is to treat this problem as a missing value problem as follows:

Suppose with each x_i we also observe the corresponding value of N , say, m_i . Therefore, the complete observation becomes

$$\mathcal{D}_{1c} = \{(x_1, m_1), \dots, (x_n, m_n)\}. \quad (5.20)$$

Based on the complete data, the log-likelihood function becomes

$$l(\theta, \sigma, p) = n \ln p + \left(\sum_{i=1}^n m_i - n \right) \ln(1-p) - n \ln \sigma - \frac{1}{2\sigma^2} \sum_{i=1}^n \frac{(x_i - \theta)^2}{m_i}. \quad (5.21)$$

Hence, the ML estimates of θ, σ , and p based on the complete observation \mathcal{D}_{1c} can be obtained by maximizing (5.21). If we denote them as $\hat{\theta}_c, \hat{\sigma}_c$ and \hat{p}_c , respectively, then

$$\begin{aligned} \hat{\theta}_c &= \frac{1}{\sum_{i=1}^n \frac{1}{m_i}} \sum_{i=1}^n \frac{x_i}{m_i}, \\ \hat{\sigma}_c^2 &= \frac{1}{n} \sum_{i=1}^n \frac{(x_i - \hat{\theta}_c)^2}{m_i}, \\ \hat{p}_c &= \frac{n}{\sum_{i=1}^n m_i}. \end{aligned} \quad (5.22)$$

Hence, in this case, the EM algorithm takes the following form. Suppose at the k th stage of the EM algorithm the estimates of θ, σ , and p are $\theta^{(k)}, \sigma^{(k)}$, and $p^{(k)}$, respectively. Then at the ‘‘E’’-step of the EM algorithm, the ‘‘pseudo’’ log-likelihood function can be written as

$$l_s(\theta, \sigma, p | \theta^{(k)}, \sigma^{(k)}, p^{(k)}) = n \ln p + \left(\sum_{i=1}^n a_i^{(k)} - n \right) \ln(1-p) - n \ln \sigma - \frac{1}{2\sigma^2} \sum_{i=1}^n b_i^{(k)} (x_i - \theta)^2. \quad (5.23)$$

Here,

$$\begin{aligned} a_i^{(k)} &= E(N | Y = x_i, \theta^{(k)}, \sigma^{(k)}, p^{(k)}) \\ &= \frac{\sum_{n=1}^{\infty} \sqrt{n} (1-p^{(k)})^{n-1} e^{-\frac{(x_i - \theta^{(k)})^2}{2n(\sigma^{(k)})^2}}}{\sum_{j=1}^{\infty} (1-p^{(k)})^{j-1} e^{-\frac{(x_i - \theta^{(k)})^2}{2j(\sigma^{(k)})^2}} / \sqrt{j}} \end{aligned} \quad (5.24)$$

and

$$\begin{aligned} b_i^{(k)} &= E(1/N | Y = x_i, \theta^{(k)}, \sigma^{(k)}, p^{(k)}) \\ &= \frac{\sum_{n=1}^{\infty} n^{-3/2} (1-p^{(k)})^{n-1} e^{-\frac{(x_i - \theta^{(k)})^2}{2n(\sigma^{(k)})^2}}}{\sum_{j=1}^{\infty} (1-p^{(k)})^{j-1} e^{-\frac{(x_i - \theta^{(k)})^2}{2j(\sigma^{(k)})^2}} / \sqrt{j}}. \end{aligned} \quad (5.25)$$

Therefore, at the “M”th step, the maximization of the “pseudo” log-likelihood functions provides

$$\begin{aligned}\theta^{(k+1)} &= \frac{\sum_{i=1}^n b_i^{(k)} x_i}{\sum_{j=1}^n b_j^{(k)}}, \quad \sigma^{(k+1)} = \frac{1}{n} \sum_{i=1}^n b_i^{(k)} (x_i - \theta^{(k+1)})^2, \\ p^{(k+1)} &= \frac{n}{\sum_{i=1}^n a_i^{(k)}}.\end{aligned}\quad (5.26)$$

Note that to start the EM algorithm, we need some initial estimates of θ , σ , and p . We suggest the following. Use the sample mean and the sample standard deviation as the initial estimates of θ and σ , respectively. We can start the EM algorithm for different initial values of p from $(0, 1)$. The EM algorithm can be explicitly written as follows:

EM ALGORITHM

Step 1: Choose some initial estimates of θ , σ , and p , say, $\theta^{(1)}$, $\sigma^{(1)}$, and $p^{(1)}$ as described above. Put $k = 1$.

Step 2: Compute $a_i^{(k)}$ and $b_i^{(k)}$ as in (5.24) and (5.25), respectively.

Step 3: Compute $\theta^{(k+1)}$, $\sigma^{(k+1)}$, and $p^{(k+1)}$ as in (5.26).

Step 4: Check the convergence. If the convergence criterion is satisfied, then stop; otherwise, put $k = k + 1$, and go to Step 2.

The observed Fisher information matrix also can be easily obtained as the last step of the corresponding algorithm. The observed Fisher information matrix can be written as $\mathbf{I} = \mathbf{B} - \mathbf{S}\mathbf{S}^\top$. Here, \mathbf{B} is a 3×3 Hessian matrix of the “pseudo” log-likelihood function (5.23), and \mathbf{S} is the corresponding gradient vector. If we denote $\mathbf{B} = ((b_{ij}))$ and $\mathbf{S} = (s_i)$, then at the k th stage of the EM algorithm,

$$b_{11} = \frac{1}{(\sigma^{(k)})^2} \sum_{i=1}^n b_i^{(k)}, \quad b_{12} = b_{21} = \frac{2}{(\sigma^{(k)})^3} \sum_{i=1}^n b_i^{(k)} (x_i - \theta^{(k)}),$$

$$b_{22} = -\frac{n}{(\sigma^{(k)})^2} + \frac{3}{(\sigma^{(k)})^4} \sum_{i=1}^n b_i^{(k)} (x_i - \theta^{(k)})^2$$

$$b_{33} = \frac{n}{(p^{(k)})^2} + \frac{\sum_{i=1}^n a_i^{(k)} - n}{(1 - p^{(k)})^2}, \quad b_{13} = b_{31} = b_{23} = b_{32} = 0.$$

$$\begin{aligned}s_1 &= \frac{1}{(\sigma^{(k)})^2} \sum_{i=1}^n b_i^{(k)} (x_i - \theta^{(k)}), \quad s_2 = -\frac{n}{\sigma^{(k)}} + \frac{1}{(\sigma^{(k)})^3} \\ &\times \sum_{i=1}^n b_i^{(k)} (x_i - \theta^{(k)})^2, \quad s_3 = \frac{n}{p^{(k)}} - \frac{\sum_{i=1}^n a_i^{(k)}}{1 - p^{(k)}}.\end{aligned}$$

Therefore, if $\mathbf{I}^{-1} = ((f_{ij}))$, then $100(1 - \alpha)\%$ confidence intervals of θ , σ , and p can be obtained as

$$\begin{aligned}(\widehat{\theta} - z_{\alpha/2} f^{11}, \widehat{\theta} + z_{\alpha/2} f^{11}), \quad (\widehat{\sigma} - z_{\alpha/2} f^{22}, \widehat{\sigma} + z_{\alpha/2} f^{22}), \\ (\widehat{p} - z_{\alpha/2} f^{33}, \widehat{p} + z_{\alpha/2} f^{33}),\end{aligned}$$

respectively. Here z_α denotes the α th percentile point of a standard normal distribution.

5.4.3 Testing of Hypothesis

In this section, we discuss some testing of hypotheses problems which have some practical importance.

Problem 1 We want to test

$$H_0 : \theta = \theta_0 \quad \text{vs.} \quad H_1 : \theta \neq \theta_0. \quad (5.27)$$

The problem is of interest as it tests whether the distribution has a specific mean or not. We propose to use the likelihood ratio test (LRT) for this purpose. To compute the LRT statistic, we need to compute the ML estimates of σ and p , when $\theta = \theta_0$. In this case also we can use the same EM algorithm with the obvious modification that at each stage $\theta^{(k)}$ is replaced by θ_0 . Therefore, if $\widehat{\theta}$, $\widehat{\sigma}$, and \widehat{p} denote the ML estimates of θ , σ , and p , respectively, without any restriction, and $\widetilde{\sigma}$ and \widetilde{p} denote the ML estimators of σ and p , respectively, under H_0 , then

$$2(l(\widehat{\theta}, \widehat{\sigma}, \widehat{p}) - l(\theta_0, \widetilde{\sigma}, \widetilde{p})) \longrightarrow \chi_1^2.$$

Problem 2 We want to test

$$H_0 : p = 1 \quad \text{vs.} \quad H_1 : p < 1. \quad (5.28)$$

The problem is of interest as it tests whether the distribution is normal or not. In this case under H_0 , the ML estimates of θ and σ can be obtained as

$$\widetilde{\theta} = \frac{\sum_{i=1}^n x_i}{n} \quad \text{and} \quad \widetilde{\sigma} = \sqrt{\frac{\sum_{i=1}^n (x_i - \widetilde{\theta})^2}{n}}.$$

In this case p is in the boundary under H_0 ; hence, the standard results do not hold. But using Theorem 3 of Self and Liang [28], it follows that

$$2(l(\widehat{\theta}, \widehat{\sigma}, \widehat{p}) - l(\widetilde{\theta}, \widetilde{\sigma}, 1)) \longrightarrow \frac{1}{2} + \frac{1}{2} \chi_1^2.$$

5.5 SGSN Regression Model

In this section, we introduce the following regression model:

$$\mathbf{Y} = \mathbf{X}^\top \boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad (5.29)$$

where \mathbf{Y} is a $n \times 1$ observed vector, $\mathbf{X} = (X_1, X_2, \dots, X_k)^\top$ is a k -dimensional covariate vector, $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_k)^\top$ is a k -vector of regression coefficients, \mathbf{a}^\top denotes the transpose of an arbitrary vector \mathbf{a} , and $\epsilon \sim \text{SGSN}(\sigma, p)$. Note that it is a standard multiple linear regression model when ϵ has a normal distribution with mean zero and finite variance. Since the normal distribution is a special case of the proposed SGSN distribution, therefore, the model (5.29) will be more flexible than the standard multiple linear regression model.

The main aim of this section is to discuss the estimation of the unknown parameters of the model (5.29). Let

$$\mathcal{D} = \{(y_1, \mathbf{x}_1), (y_2, \mathbf{x}_2), \dots, (y_n, \mathbf{x}_n)\},$$

be n independent observations from the model (5.29); the problem is to estimate the unknown parameters $\beta_1, \dots, \beta_k, \sigma, p$. We use the following notation $\mathbf{x}_i = (x_{1i}, \dots, x_{ki})^\top$, $i = 1, \dots, n$.

Let us try to compute the ML estimates of the unknown parameters. The log-likelihood function of the observed data \mathcal{D} based on the model (5.29) can be obtained as

$$l(\boldsymbol{\theta}) = n \ln p - n \ln(1-p) - n \ln \sigma + \sum_{i=1}^n \ln \left(\sum_{j=1}^{\infty} \frac{(1-p)^j}{\sqrt{j}} \phi \left(\frac{y_i - \mathbf{x}_i^\top \boldsymbol{\beta}}{\sigma \sqrt{j}} \right) \right), \quad (5.30)$$

where $\boldsymbol{\theta} = (\boldsymbol{\beta}, \sigma, p)^\top$. Therefore, the ML estimates of the unknown parameters can be obtained by maximizing the log-likelihood function (5.30) with respect to the unknown parameters. Taking derivatives with respect to the unknown parameters, the normal equations become

$$\begin{aligned} \dot{l}_{\beta_1} &= \frac{\partial}{\partial \beta_1} l(\boldsymbol{\theta}) = 0, \dots, \dot{l}_{\beta_k} = \frac{\partial}{\partial \beta_k} l(\boldsymbol{\theta}) = 0, \\ \dot{l}_{\sigma} &= \frac{\partial}{\partial \sigma} l(\boldsymbol{\theta}) = 0, \quad \dot{l}_p = \frac{\partial}{\partial p} l(\boldsymbol{\theta}) = 0. \end{aligned} \quad (5.31)$$

Clearly, they cannot be solved explicitly. One needs to solve $k+2$ nonlinear equations simultaneously to compute the ML estimates of the unknown parameters. To avoid that problem, we propose to use this problem as a missing value problem and provide an efficient EM algorithm to compute the ML estimates of the unknown parameters. The main idea is as follows. Suppose along with (y_i, \mathbf{x}_i) we also observe m_i , where m_i denotes the value of N in this case, for $i = 1, 2, \dots, n$. Therefore, the complete data will be of the form

$$\mathcal{D}^c = \{(y_1, \mathbf{x}_1, m_1), \dots, (y_n, \mathbf{x}_n, m_n)\}. \quad (5.32)$$

First, we will show that if m_i 's are known, then the ML estimates of $\boldsymbol{\beta}$, σ , and p can be obtained in explicit forms.

Based on the complete data (5.32), the log-likelihood function without the additive constant becomes

$$l_c(\boldsymbol{\beta}, \sigma, p) = n \ln p + \left(\sum_{i=1}^n m_i - n \right) \ln(1-p) - n \ln \sigma - \frac{1}{2\sigma^2} \sum_{i=1}^n \frac{1}{m_i} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2. \quad (5.33)$$

It can be easily seen that $l_c(\boldsymbol{\beta}, \sigma, p)$ as given in (5.33) is a unimodal function. If $\hat{\boldsymbol{\beta}}_c$, $\hat{\sigma}_c$, and \hat{p} maximize $l_c(\boldsymbol{\beta}, \sigma, p)$, then they can be obtained as

$$\begin{aligned} \hat{\boldsymbol{\beta}}_c &= \left[\sum_{i=1}^n \frac{1}{m_i} \mathbf{x}_i \mathbf{x}_i^\top \right]^{-1}, \quad \hat{\sigma}_c^2 = \frac{1}{n} \sum_{i=1}^n \frac{1}{m_i} (y_i - \mathbf{x}_i^\top \hat{\boldsymbol{\beta}}_c)^2, \\ \hat{p} &= \frac{n}{\sum_{i=1}^n m_i}. \end{aligned} \quad (5.34)$$

Therefore, it is clear that if m_i 's are known, then the ML estimates of $\boldsymbol{\beta}$, σ , and p can be obtained quite conveniently, and one does not need to solve any nonlinear equation. This is the main motivation of the proposed EM algorithm. Now we are ready to provide the EM algorithm for this problem. We will show how to move from the r th step to the $(r+1)$ th step of the EM algorithm.

We will use the following notations. Let us denote $\boldsymbol{\beta}^{(r)}$, $\sigma^{(r)}$, and $p^{(r)}$ as the estimates of $\boldsymbol{\beta}$, σ , and p , respectively, for $i = 1, 2, \dots, n$, at the r th iteration of the EM algorithm. Then at the "E"-step of the EM algorithm, the pseudo log-likelihood function becomes

$$\begin{aligned} l_s(\boldsymbol{\beta}, \sigma, p | \boldsymbol{\beta}^{(r)}, \sigma^{(r)}, p^{(r)}) &= n \ln p + \left(\sum_{i=1}^n c_i^{(r)} - n \right) \ln(1-p) \\ &\quad - n \ln \sigma - \frac{1}{2\sigma^2} \sum_{i=1}^n d_i^{(r)} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2, \end{aligned} \quad (5.35)$$

where

$$\begin{aligned} c_i^{(r)} &= E(N | y_i, \mathbf{x}_i, \boldsymbol{\beta}^{(r)}, \sigma^{(r)}, p^{(r)}) \\ &= \frac{\sum_{n=1}^{\infty} \sqrt{n} (1-p^{(r)})^{n-1} e^{-\frac{(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r)})^2}{2n(\sigma^{(r)})^2}}}{\sum_{j=1}^{\infty} (1-p^{(r)})^{j-1} e^{-\frac{(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r)})^2}{2j(\sigma^{(r)})^2}} / \sqrt{j}} \end{aligned} \quad (5.36)$$

and

$$\begin{aligned} d_i^{(r)} &= E(1/N | y_i, \mathbf{x}_i, \boldsymbol{\beta}^{(r)}, \sigma^{(r)}, p^{(r)}) \\ &= \frac{\sum_{n=1}^{\infty} n^{-3/2} (1-p^{(r)})^{n-1} e^{-\frac{(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r)})^2}{2n(\sigma^{(r)})^2}}}{\sum_{j=1}^{\infty} (1-p^{(r)})^{j-1} e^{-\frac{(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r)})^2}{2j(\sigma^{(r)})^2}} / \sqrt{j}}. \end{aligned} \quad (5.37)$$

Therefore, at the “M”th step, the maximization of the “pseudo” log-likelihood function provides

$$\begin{aligned}\boldsymbol{\beta}^{(r+1)} &= \left[\sum_{i=1}^n d_i^{(r)} \mathbf{x}_i \mathbf{x}_i^\top \right]^{-1}, \quad \sigma^{(r+1)} \\ &= \sqrt{\frac{1}{n} \sum_{i=1}^n d_i^{(r)} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r+1)})^2}, \quad \hat{p} = \frac{n}{\sum_{i=1}^n c_i^{(r)}};\end{aligned}\quad (5.38)$$

moreover, these are unique solutions.

Now to start the EM algorithm, we suggest the following initial guesses. Use ordinary least squares estimates of $\boldsymbol{\beta}$ as the initial estimate of $\boldsymbol{\beta}$ and the corresponding square root of the residual sums of squares as the initial estimate of σ . As before, we can start the EM algorithm with different initial values of $p \in (0, 1)$. The EM algorithm can be written as follows.

EM ALGORITHM

Step 1: Choose some initial estimates of $\boldsymbol{\beta}$, σ , and p , say, $\boldsymbol{\beta}^{(1)}$, $\sigma^{(1)}$, and $p^{(1)}$ as described above. Put $r = 1$.

Step 2: Compute $c_i^{(r)}$ and $\hat{c}_i^{(r)}$ as in (5.36) and (5.37), respectively.

Step 3: Compute $\boldsymbol{\beta}^{(r+1)}$, $\sigma^{(r+1)}$, and $p^{(r+1)}$ as in (5.38).

Step 4: Check the convergence. If the convergence criterion is satisfied, then stop; otherwise, put $r = r + 1$, and go to Step 2.

In this case also the observed Fisher information matrix also can be obtained as before, in the last step of the EM algorithm. The observed Fisher information matrix can be written as $\tilde{\mathbf{I}} = \tilde{\mathbf{B}} - \tilde{\mathbf{S}}\tilde{\mathbf{S}}^\top$. Here, $\tilde{\mathbf{B}}$ is a $(k+3) \times (k+3)$ Hessian matrix of the “pseudo” log-likelihood function (5.35), and $\tilde{\mathbf{S}}$ is the corresponding gradient vector. If we denote $\tilde{\mathbf{B}} = ((\tilde{b}_{ij}))$ and $\tilde{\mathbf{S}} = (\tilde{s}_i)$, then at the r th stage of the EM algorithm, for $j, l = 1, \dots, k$ and for $j \neq l$,

$$\tilde{b}_{jj} = \frac{1}{(\sigma^{(r)})^2} \sum_{i=1}^n d_i^{(r)} x_{ji}^2, \quad \tilde{b}_{jl} = \tilde{b}_{lj} = \frac{1}{(\sigma^{(r)})^3} \sum_{i=1}^n d_i^{(r)} x_{ji} x_{li},$$

$$\tilde{b}_{(k+1)(k+1)} = -\frac{n}{(\sigma^{(r)})^2} + \frac{3}{(\sigma^{(r)})^4} \sum_{i=1}^n d_i^{(k)} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r)})^2,$$

$$\tilde{b}_{j(k+1)} = \tilde{b}_{(k+1)j} = \frac{2}{(\sigma^{(r)})^3} \sum_{i=1}^n d_i^{(k)} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r)}) x_{ji},$$

$$\tilde{b}_{(k+2)(k+2)} = \frac{n}{(p^{(r)})^2} + \frac{\sum_{i=1}^n c_i^{(r)} - n}{(1-p^{(r)})^2},$$

$$b_{j(k+2)} = 0; \quad j = 1, \dots, k+1.$$

$$s_j = \frac{1}{(\sigma^{(r)})^2} \sum_{i=1}^n d_i^{(r)} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r)}) x_{ji}; \quad j = 1, \dots, k,$$

$$s_{(k+1)} = -\frac{n}{\sigma^{(r)}} + \frac{1}{(\sigma^{(r)})^3} \sum_{i=1}^n d_i^{(r)} (y_i - \mathbf{x}_i^\top \boldsymbol{\beta}^{(r)})^2,$$

$$s_{(k+2)} = \frac{n}{p^{(r)}} - \frac{\sum_{i=1}^n c_i^{(r)} - n}{1-p^{(r)}}.$$

Therefore, if $\tilde{\mathbf{I}}^{-1} = ((\tilde{f}_{ij}))$, then $100(1 - \alpha)\%$ confidence intervals of β_j , σ , and p can be obtained as

$$\begin{aligned}(\hat{\beta}_j - z_{\alpha/2} \tilde{f}_{jj}, \hat{\beta}_j + z_{\alpha/2} \tilde{f}_{jj}); \quad j = 1, \dots, k, \\ (\hat{\sigma} - z_{\alpha/2} \tilde{f}_{(k+1)(k+1)}, \hat{\sigma} + z_{\alpha/2} \tilde{f}_{(k+1)(k+1)}), \\ (\hat{p} - z_{\alpha/2} \tilde{f}_{(k+2)(k+2)}, \hat{p} + z_{\alpha/2} \tilde{f}_{(k+2)(k+2)}),\end{aligned}$$

respectively. Here z_{α} denotes the α th percentile point of a standard normal distribution.

5.6 Simulation Results

In this section, we present some simulation results both for the LS-SGSN and SGSN regression models. The main idea is to see how the proposed EM algorithms behave in these cases. All the computations are performed using R software, and it can be obtained from the corresponding author on request.

5.6.1 LS-SGSN Model

In this section, we have generated samples from a LS-SGSN distribution for different sample sizes n and different p values. We have kept the values of θ and σ to be the same, namely, $\theta = 0$ and $\sigma = 1$. We have taken $n = 25, 50, 75$, and 100 and $p = 0.2, 0.4, 0.6$, and 0.8 . We have used the EM algorithm as it has been described in Sect. 5.3. In all the cases, we have used the true value as the initial guesses. We stop the iteration when the absolute difference of the consecutive estimates is less than 10^{-6} , for all the unknown parameters. We have reported the average ML estimates of θ , σ , p , and the associated mean squared errors (MSEs) based on 1000 replications. The results are reported in Table 5.1.

Some of the points are quite clear from the above experimental results. It is observed that ML estimates of θ , σ , and p provide unbiased estimates of the corresponding parameters. As sample size increases, in all the cases considered, the MSEs decrease. It indicates consistency property of the ML estimates. The EM algorithm converges in all the cases. Hence, the EM algorithm works well in this case.

Table 5.1 ML estimates of θ , σ , and p and the associated MSEs (reported below within brackets) for different sample sizes and for different p values

n		$p = 0.2$	$p = 0.4$	$p = 0.6$	$p = 0.8$
25	θ	0.0003 (0.1230)	0.0062 (0.0738)	0.0007 (0.0621)	-0.0026 (0.0483)
	σ	0.9251 (0.0471)	0.9177 (0.0419)	0.9293 (0.0369)	0.9314 (0.0374)
	p	0.1883 (0.0224)	0.3703 (0.0116)	0.5699 (0.0103)	0.7546 (0.0081)
50	θ	0.0094 (0.0651)	-0.0004 (0.0398)	-0.0030 (0.0320)	0.0067 (0.0243)
	σ	0.9515 (0.0396)	0.9427 (0.0326)	0.9519 (0.0294)	0.9569 (0.0211)
	p	0.1890 (0.0121)	0.3792 (0.0099)	0.5768 (0.0089)	0.7748 (0.0075)
75	θ	0.0104 (0.0432)	0.0036 (0.0271)	0.0060 (0.0189)	0.0028 (0.0149)
	σ	0.9606 (0.0273)	0.9747 (0.0250)	0.9720 (0.0231)	0.9683 (0.0200)
	p	0.1891 (0.0099)	0.3835 (0.0079)	0.5775 (0.0056)	0.7787 (0.0047)
100	θ	-0.0004 (0.0318)	-0.0018 (0.0196)	-0.0023 (0.0144)	0.0016 (0.0115)
	σ	0.9805 (0.0165)	0.9767 (0.0234)	0.9889 (0.0193)	0.9839 (0.0165)
	p	0.1948 (0.0071)	0.3927 (0.0056)	0.5860 (0.0032)	0.7880 (0.0027)

Table 5.2 ML estimates of β_1 , β_2 , and σ and the associated MSEs (reported below within brackets) for different sample sizes and for different p values

n		$p = 0.2$	$p = 0.4$	$p = 0.6$	$p = 0.8$
25	β_1	1.0063 (0.1348)	1.0044 (0.0824)	0.9998 (0.0612)	0.9979 (0.0487)
	β_2	2.0006 (0.0952)	1.9964 (0.0568)	2.0010 (0.0485)	2.0062 (0.0400)
	σ	0.9274 (0.0264)	0.9286 (0.0243)	0.9356 (0.0238)	0.9398 (0.0229)
	p	0.1966 (0.0014)	0.3878 (0.0061)	0.5729 (0.0173)	0.7714 (0.0207)
50	β_1	1.0043 (0.0502)	1.0008 (0.0316)	0.9987 (0.0227)	1.0013 (0.0197)
	β_2	2.0011 (0.0703)	1.9995 (0.0427)	2.0001 (0.0331)	2.0058 (0.0269)
	σ	0.9508 (0.0173)	0.9323 (0.0211)	0.9389 (0.0212)	0.9428 (0.0199)
	p	0.1987 (0.0008)	0.3889 (0.0048)	0.5787 (0.0143)	0.7899 (0.0178)
75	β_1	0.9917 (0.0422)	1.0001 (0.0217)	0.9997 (0.0208)	1.0001 (0.0146)
	β_2	2.0005 (0.0367)	2.0005 (0.0226)	2.0003 (0.0193)	2.0041 (0.0149)
	σ	0.9615 (0.0165)	0.9576 (0.0142)	0.9545 (0.0128)	0.9758 (0.0111)
	p	0.1998 (0.0004)	0.3976 (0.0028)	0.5889 (0.0013)	0.7987 (0.0011)
100	β_1	0.9993 (0.0242)	1.0021 (0.0163)	1.0016 (0.0127)	0.9997 (0.0095)
	β_2	1.9992 (0.0261)	1.9987 (0.0147)	1.9970 (0.0126)	2.0002 (0.0100)
	σ	0.9893 (0.0056)	0.9892 (0.0058)	0.9888 (0.0059)	0.9917 (0.0056)
	p	0.2001 (0.0001)	0.3999 (0.0016)	0.5998 (0.0008)	0.7999 (0.0005)

5.6.2 SGSN Regression Model

In this section, we have performed some simulation experiments for SGSN regression model. Here, we have taken $k = 2$, $\beta_1 = 1$, $\beta_2 = 2$, and $\sigma = 1$. We have varied $n = 25, 50, 75$, and 100 and $p = 0.2, 0.4, 0.6$, and 0.8 . We have generated the entries of the \mathbf{X} matrix from i.i.d. normal random variables, with mean zero and variance one. For a fixed, n , the generated \mathbf{X} matrix remains fixed. The error random variables are generated from a SGSN(σ, p) distribution. For each n, σ , and p , we generate Y . We compute the ML estimates of β_1, β_2, σ , and p , based on the EM algorithm as described in Sect. 5.4. In all the cases, we report the average ML estimates and the associated MSEs based on 1000 replications. The results are reported in Table 5.2.

Some of the points are quite clear from both the tables. It is observed that even for the SGSN regression model, the proposed EM algorithm works quite well in all the cases considered. The ML estimates provide consistent estimates of the corresponding unknown parameters, and the EM algorithm also converges in each case.

5.7 Real Data Analysis

5.7.1 LS-SGSN Model

In this section, we present the analysis of one data set based on LS-SGSN model mainly for illustrative purposes. The data set has been obtained from Lawless [21] (page 228). It arose from test on the endurance of deep groove ball bearings. It represents the number of million revolutions before failure for each of the 23 ball bearings in the life test, and they are as follows:

17.88 28.92 33.00 41.52 42.12 45.60 48.40 51.84
51.96 54.12 55.56 67.80 68.64 68.64 68.88 84.12 93.12
98.64 105.12 105.84 127.92 128.04 173.40.

The mean and standard deviation of the data points are 72.2 and 36.7, respectively. We have used the EM algorithm with the initial estimates of θ , σ , and p as 72.2, 36.7, and 0.5, respectively. We stop the iteration when the differences between the consecutive estimates are less than 10^{-4} for all the three parameters. The ML estimates and the associated log-likelihood (ll) value are as follows:

$$\hat{\theta} = 68.443, \quad \hat{\sigma} = 26.088 \quad \hat{p} = 0.554, \quad ll = -127.4590.$$

The corresponding 95% confidence intervals of θ , σ , and p are obtained as

$$(56.513, 80.373), \quad (18.857, 33.320), \quad (0.409, 0.699),$$

respectively. To check the effect of the initial guess to the final estimates, we have used different other initial guesses, mainly the p values ranging from 0.1 to 0.9, but in all the cases it converges to the same estimate. It confirms that the initial estimates do not affect the convergence of the EM algorithm.

5.7.2 SGSN Regression Model

In this section, we have analyzed one rocket propellant data set from Montgomery et al. [23] (Chapter 2). The data represent the shear strength (Y) and the age of the propellant (X). Here the shear strength is in psi, and the age is in weeks. The observations are given below in the format (Y, X) :

(2158.70, 15.50) (1678.15, 23.75) (2316.00, 8.00)
 (2061.30, 17.00) (2207.50, 5.50) (1708.30, 19.00) (1784.70,
 24.00) (2575.00, 2.50) (2357.90, 7.50) (2256.70, 11.00)
 (2165.20, 13.00) (2399.55, 3.75) (1779.80, 25.00) (2336.75,
 9.75) (1765.30, 22.00) (2053.50, 18.00) (2414.40, 6.00)
 (2200.50, 12.50) (2654.20, 2.00) (1753.70, 21.50).

We have used the following model to analyze the data:

$$Y = \beta_1 + \beta_2 X + \epsilon.$$

Here ϵ is assumed to follow a GSSN distribution with mean zero and finite variance. Assuming that ϵ follows normal distribution, we obtain the least squares estimates of β_1 , β_2 , and σ as 2627.82, -37.15 , and 91.17, respectively. We have used these estimates as the initial estimates, and we have used the initial estimate of p as 0.5. Using the EM algorithm, we obtain the ML estimates of the unknown parameters, and the corresponding log-likelihood values are as follows:

$$\hat{\beta}_1 = 2649.58, \quad \hat{\beta}_2 = -37.58, \quad \hat{\sigma} = 53.42,$$

$$\hat{p} = 0.34, \quad ll = -88.97.$$

The associated 95% confidence intervals are as follows:

$$(2584.95, 2714.18), \quad (-41.83, -33.34),$$

$$(36.87, 69.98), \quad (0.22, 0.46).$$

5.8 Conclusions

In this paper, we have considered LS-SGSN distribution. It is symmetric, and it has three parameters. Hence, it is more flexible than the normal distribution. The normal distribution can be obtained as a special case of the LS-SGSN distribution. We have proposed a very efficient EM algorithm, and it is observed that the EM algorithm works quite well. We have further considered SGSN regression model, which is more flexible than the standard Gaussian regression model. In this case also we have proposed a very efficient EM algorithm, and the performance of the proposed EM algorithm is quite satisfactory.

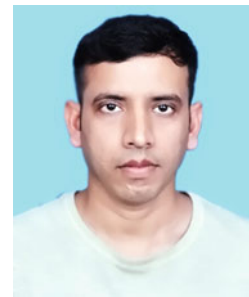
References

1. Arnold, B.C., Beaver, R.J.: Hidden truncation models. *Sankhya* **62**, 23–35 (2000)
2. Arnold, B.C., Beaver, R.J., Groeneveld, R.A., Meeker, W.Q.: The non-truncated marginal of a truncated bivariate normal distribution. *Psychometrika* **58**, 471–488 (1993)
3. Azzalini, A.A.: A class of distributions which include the normal. *Scand. J. Stat.* **12**, 171–178 (1985)
4. Azzalini, A.A., Capitanio, A.: *The Skew-Normal and Related Families*. Cambridge University, Cambridge (2013)
5. Azzalini, A.A., Dalla Valle, A.: The multivariate skew normal distribution: *Biometrika* **83**, 715–726 (1996)
6. Balakrishnan, N., Kundu, D.: Birnbaum-saunders distribution: a review of models, analysis and applications (with discussions). *Appl. Stoch. Model. Bus. Ind.* **35**(1), 4–132 (2019)
7. Birnbaum, Z., Saunders, S.C.: A new family of life distributions. *J. Appl. Probab.* **6**, 319–327 (1969)
8. Cook, R.D.: Assessment of local influence. *J. R. Stat. Soc. Ser. B* **48**, 133–169 (1986)
9. Fachini, J.B., Ortega, E.M.M., Louzada-Neto, F.: Influence diagnostics for polynomial hazards in presence of covariates. *Statistical Methods and its Applications* **17**, 413–433 (2008)

10. Gupta, R.C., Gupta, R.D.: Generalized skew normal model. *TEST* **13**, 1–24 (2004)
11. Hashimoto, E.M., Ortega, E.M.M., Cancho, V.G., Cordeiro, G.M.: The log-exponentiated Weibull regression model for interval censored data. *Comput. Stat. Data Anal.* **54**, 1017–1035 (2010)
12. Hashimoto, E.M., Ortega, E.M.M., Cordeiro, G.M., Cancho, V.G.: The Poisson Birnbaum-Saunders model with long-term survivors. *Statistics* **48**(6), 1394–1413 (2014)
13. Johnson, N.L., Kotz, S., Balakrishnan, N.: Continuous univariate distribution, vol. 1. Wiley, New York (1995)
14. Kundu, D.: Geometric skewed normal distribution. *Sankhya Ser. B* **76**(Part 2), 167–189 (2014)
15. Kundu, D.: Bivariate log Birnbaum-Saunders Distribution. *Statistics* **49**(4), 900–917 (2015)
16. Kundu, D.: Bivariate sinh-normal distribution and a related model. *Brazilian Journal of Probability and Statistics* **29**(3), 590–607 (2015a)
17. Kundu, D.: Multivariate geometric skew normal distribution. *Statistics* **51**(6), 1377–1397 (2017)
18. Kundu, D., Gupta, R.D.: Power Normal Distribution. *Statistics* **47**(1), 110–125 (2013)
19. Kundu, D., Nekoukhou, V.: Univariate and bivariate geometric discrete generalized exponential distribution. *J. Stat. Theory Pract.* **12**(3), 595–614 (2018)
20. Lachos, V.H., Bolfarine, H., Arellano-Valle, R.B., Montenegro, L.C.: Likelihood-Based Inference for Multivariate Skew-Normal Regression Models. *Commun. Stat.- Theory Methods* **36**, 1769–1786 (2007)
21. Lawless, J.F.: *Statistical Models and Methods for Lifetime Data*. Wiley, New York (1982)
22. Louis, T.A.: Finding the observed information matrix when using the EM algorithm. *J. R. Stat. Soc. Series B* **44**(2), 226–233 (1982)
23. Montgomery, D.C., Peck, E.A., Vining, G.G.: *Introduction to Linear Regression Analysis*, 3rd edn, Wiley, New York (2001)
24. Ortega, E.M.M., Bolfarine, H., Paula, G.A.: Influence diagnostics in generalized log-gamma regression model. *Comput. Stat. Data Anal.* **42**, 165–186 (2003)
25. Rieck, J.R.: Statistical analysis for the Birnbaum-Saunders fatigue life distribution, Ph.D. thesis. Clemson University, Department of Mathematical Sciences, Canada (1989)
26. Rieck, J.R., Nedelman, J.R.: A log-linear model for the Birnbaum-Saunders distribution. *Technometrics* **33**, 51–60 (1991)
27. Sahu, S.K., Dey, D.K., Branco, M.D.: A new class of multivariate skew distributions with applications to Bayesian regression models. *Can. J. Stat.* **31**(2), 129–150 (2003)
28. Self, S.G., Liang, K.-L.: Asymptotic properties of the maximum likelihood estimators and likelihood ratio test under non-standard conditions. *J. Am. Stat. Assoc.* **82**, 605–610 (1987)
29. Silva, G.O., Ortega, E.M.M., Cancho, V.G., Barreto, M.L.: Log-Burr XII regression model with censored data. *Comput. Stat. Data Anal.* **52**, 3820–3842 (2008)
30. Zhang, Y., Lu, X., Desmond, A.F.: Variable selection in a log-linear Birnbaum-Saunders regression model for high-dimensional survival data via elastic-net and stochastic EM. *Technometrics* **58**(3), 383–392 (2016)



Professor Debasis Kundu received his Ph.D. from the Pennsylvania State University in 1989 under the guidance of Professor C.R. Rao. After finishing his Ph.D. he joined The University of Texas at Dallas as an Assistant Professor before joining Indian Institute of Technology Kanpur in 1990. He served as the Head of the Department of Mathematics and Statistics, IIT Kanpur from 2011 to 2014, and currently he is the Dean of Faculty Affairs at IIT Kanpur. Professor Kundu is holding an Endowed Chair Professor post of the Institute since 2008. Professor Kundu works on different areas of Statistics. His major research interests are on Distribution Theory, Lifetime Data Analysis, Censoring, Statistical Signal Processing and Statistical Computing. He has published more than 300 research papers in different refereed journals and more than 13,000 google scholar citations with H index 58. He has co-authored three research monographs and co-edited one book. He is a Fellow of the National Academy of Sciences, India, a Fellow of the Royal Statistical Society London and a Fellow of the Indian Society of Probability and Statistics. He is currently the Editor-in-Chief of the *Journal of the Indian Society of Probability and Statistics*, and in the editorial boards of *Communications in Statistics – Theory and Methods*, *Communications in Statistics – Simulation and Computation* and *Sankhya, Ser. B*. He was in the editorial boards of the *IEEE Transactions on Reliability*, *Journal of Statistical Theory and Practice* and *Journal of Distribution Theory*.



Deepak Prajapati received his Ph.D. degree from the Department of Mathematics and Statistics, Indian Institute of Technology Kanpur, Kanpur, India, in 2019. He received his Master's degree in Applied Statistics and Informatics from Indian Institute of Technology Bombay, Mumbai, India, in 2013. He was a Postdoctoral Fellow with the Indian Statistical Institute, Delhi, India. He is currently an Assistant Professor in the Decision Sciences Area at the Indian Institute of Management, Lucknow, India. His main research interests are statistical inference, reliability theory and survival analysis, and statistical quality control.



Statistical Analysis of Modern Reliability Data

6

Yueyao Wang , I-Chen Lee , Lu Lu , and Yili Hong

Contents

6.1 Introduction	106	6.6.3 Test Plan Development	119
6.1.1 Background.....	106	6.6.4 Illustration of Test Plans	119
6.1.2 Related Literature.....	106	6.7 Sequential Test Planning Considering Dual Objectives	121
6.1.3 Overview.....	106	6.7.1 Testing Criteria.....	121
6.2 Traditional Reliability Analysis	107	6.7.2 Procedure of Sequential Planning Based on Dual Objectives.....	121
6.2.1 Background and Data.....	107	6.7.3 Polymer Composites Fatigue Testing.....	122
6.2.2 Time-to-Event Models and Parameter Estimation.....	107	6.8 Concluding Remarks	125
6.2.3 Reliability Prediction.....	108	References	125
6.3 Time-to-Event Data Analysis	109		
6.3.1 Background and Data.....	109		
6.3.2 Model for Time-to-Event Data and Parameter Estimation.....	109		
6.3.3 Model for Covariates.....	110		
6.3.4 Reliability Prediction.....	110		
6.4 Degradation Data Analysis	111		
6.4.1 Background and Data.....	112		
6.4.2 Model for Degradation Paths and Parameter Estimation.....	113		
6.4.3 Model for Covariates.....	114		
6.4.4 Reliability Prediction.....	114		
6.5 Recurrent Event Data Analysis	115		
6.5.1 Background and Data.....	115		
6.5.2 The MTRP Model and Parameter Estimation.....	116		
6.5.3 Prediction for Component Events.....	117		
6.6 Sequential Test Planning of Accelerated Life Tests	117		
6.6.1 Background and Historical Data.....	117		
6.6.2 Lifetime Model.....	118		

Abstract

Reliability analysis has been using time-to-event data, degradation data, and recurrent event data, while the associated covariates tend to be simple and constant over time. Over the past years, we have witnessed rapid development of sensor and wireless technology, which enables us to track the product usage and use environment. Nowadays, we are able to collect richer information on covariates which provides opportunities for better reliability predictions. In this chapter, we first review recent development on statistical methods for reliability analysis. We then focus on introducing several specific methods that were developed for different types of reliability data by utilizing the covariate information. Illustrations of those methods are also provided using examples from industry. We also provide a brief review on recent developments of test planning and then focus on illustrating the sequential Bayesian designs with examples of fatigue testing for polymer composites. The chapter is concluded with some discussions and remarks.

Keywords

Degradation data · Dynamic covariates · Lifetime data · Recurrent event data · Reliability prediction · Sequential test planning

Y. Wang · Y. Hong (✉)
 Department of Statistics, Virginia Tech, Blacksburg, VA, USA
 e-mail: yueyao94@vt.edu; yilihong@vt.edu

I-C. Lee
 Department of Statistics, National Cheng Kung University, Tainan, Taiwan
 e-mail: iclee@mail.ncku.edu.tw

L. Lu
 Department of Mathematics and Statistics, University of South Florida, Tampa, FL, USA
 e-mail: lulu1@usf.edu

6.1 Introduction

6.1.1 Background

Traditional reliability data analysis mainly uses time-to-event data, degradation data, and recurrent event data to make reliability predictions [1]. The covariate information involved in the reliability analysis is usually time-invariant, and the number of covariates is typically small. For time-to-event data, parametric models such as the Weibull and lognormal distributions are popular, and accelerated failure-time models are often used to incorporate covariate information on accelerating factors. For degradation data, the general path models and stochastic models are the common choices, and the covariate information is often incorporated through regression type of models. The recurrent event data are often modeled by the event intensity models or mean cumulative functions with regression type of models that are often used to incorporate covariates.

With technological advances, new information on covariates becomes available. Products and systems can be equipped with sensors and smart chips to keep track of various information on the field usage of product units, number of transfers, and environmental conditions such as temperature and humidity. Such covariate information often changes over time, so we refer to them as dynamic covariate information. Because the dynamic covariates often come in large volume and variety, it presents big data opportunities and challenges in the area of reliability analysis (e.g., [2] and [3]). Dynamic covariate data can be used for modeling and prediction of reliability because units under heavy usage often fail sooner than those lightly used. In recent years, more statistical methods for dynamic covariates have been developed to make use of this new type of covariate data.

Another important area of reliability analysis is about test planning, which focuses on how to efficiently collect various types of data to make better predictions of reliability. For accelerated life tests (ALTs), it is especially challenging to timely collect sufficient failure data because the data collection is a time-consuming process and often requires using expensive equipment for testing units under elevated stress conditions. In some laboratories, there is typically one or two machines available for testing certain material. In this case, it is impractical to test multiple samples simultaneously and therefore limits the total obtainable sample size. Another challenge with traditional test planning is that it typically relies on a single set of best guess of parameter values, which may lead to suboptimal designs when the specified parameter values are not accurate. Due to these challenges, sequential designs become popular where earlier test results can be utilized to determine the test conditions for later runs. In addition, Bayesian methods can be used to leverage prior information from the expert's knowledge or related historical

data to inform the test planning. The objective of this chapter is to review current development and then introduce the statistical methods for dynamic covariates and sequential Bayesian designs (SBDs) for ALT.

6.1.2 Related Literature

In lifetime data analysis, product usage information has been used to improve reliability models. Lawless et al. [4] consider warranty prediction problem using product usage information on return units. Constant usage information is used in [5] and [6]. Averaged product use-rate information is used in [7]. Nelson [8] and Voiculescu et al. [9] use a cumulative exposure model in ALT and reliability analysis. Hong and Meeker [10] use a cumulative exposure model to incorporate dynamic covariates and apply it to the Product D2 application.

In degradation data analysis, stochastic process models are widely used. The Wiener process [11–13], gamma process [14], and inverse Gaussian process [15, 16] are among popular models in this class. The general path models are also widely used, which include [17–21]. For accelerated destructive degradation tests, typical work includes [22–24]. Hong et al. [25] and Xu et al. [26] develop degradation models using the general path model framework to incorporate dynamic covariate information.

For recurrent event data, the common models are nonhomogeneous Poisson process (NHPP) and the renewal process (RP) (e.g., [27, 28]). Kijima [29] introduce virtual age models which can model imperfect repairs. Pham and Wang [30] develop a quasi-renewal process, and Doyen and Gaudoin [31] propose models for imperfect repairs. The trend-renewal process (TRP) proposed in [32] is applied in [33–35] and other places. Xu et al. [36] develop a multilevel trend-renewal process (MTRP) model for recurrent event with dynamic covariates.

For test planning, the optimum designs in traditional test planning framework are developed using non-Bayesian approaches (e.g., [37, 38]), and the true parameters are assumed to be known. Bayesian test planning for life data is developed in [39–41]. King et al. [42] develop optimum test plans for fatigue tests for polymer composites. Lee et al. [43] develop SBD test planning for polymer composites, and Lu et al. [44] extend it to test planning with dual objectives.

6.1.3 Overview

For the rest of this chapter, Sect. 6.2 gives an example of traditional reliability analysis. Section 6.3 describes an application on time-to-event data with dynamic covariates. Section 6.4 illustrates modeling of degradation with dynamic covariates. Section 6.5 describes the MTRP model

for describing recurrent event data with dynamic covariates. Section 6.6 introduces SBD strategies for ALTs. Section 6.7 considers sequential test planning with dual objectives. Section 6.8 contains some concluding remarks.

6.2 Traditional Reliability Analysis

In this section, we give one example on how traditional reliability analysis is typically carried out. The time-to-event data, degradation data, and recurrent event data are typically used for reliability models, and then one can use such models to make reliability predictions [1]. In traditional reliability datasets, the covariate information involved in the reliability analysis is usually simple (i.e., the covariate is time-invariant or the number of covariates is small). Here we use one example from time-to-event analysis as an illustration. Some other examples on degradation analysis can be found in [45]. This section is mainly based on [46].

6.2.1 Background and Data

Hong, Meeker, and McCalley [46] present the modeling of the lifetime data for power transformers. High-voltage power transmission transformers are important components in the electrical transmission, which has more than 150,000 units in service in the USA. The energy industry is very much interested in the prediction of the remaining life of transformers. Such prediction is typically based on historical lifetime data collected from the transformers in the field.

The paper [46] describes the modeling and analysis of lifetime data of power transformers collected by an energy company. The main goal is to make prediction for the remaining life of transformers that are still in service. Figure 6.1 shows the event plot for a subset of the power transformer lifetime data, on which the numbers on the left show the counts of transformers for each line. The truncation age is also marked on the line. The lifetime data involve censoring and truncation. Transformers that are still in service at the data freeze date (DFD) are considered to be right censored. There is information on transformers that were installed before January 1, 1980, and then subsequently failed after January 1, 1980. There is no information on transformers installed and also failed before 1980. Hence, units that were installed before 1980 are considered to be samples generated from left truncated distribution(s).

6.2.2 Time-to-Event Models and Parameter Estimation

The first step in lifetime data analysis is to specify a lifetime distribution that is suitable for describing the data. The

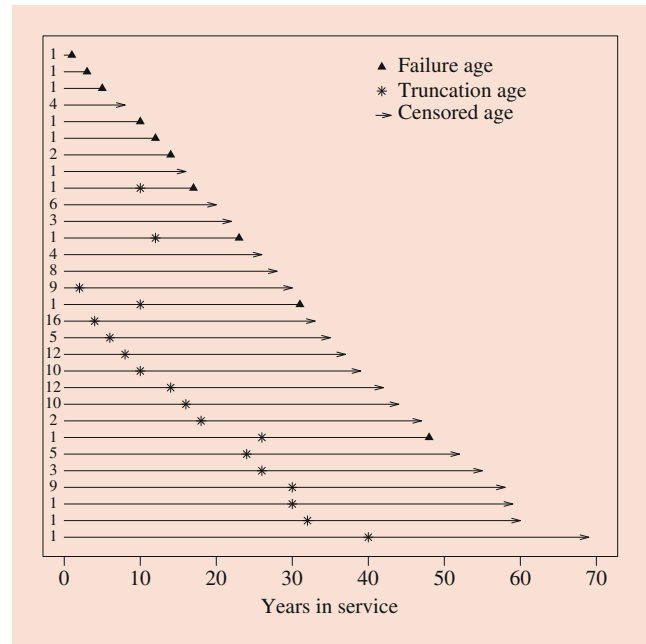


Fig. 6.1 Event plot for a subset of the power transformer lifetime data. The numbers in the left show the counts for each line

log-location-scale family of distributions is widely used in the literature. The Weibull and lognormal are widely used members in this family of distributions. Here we briefly introduce the formulae for the Weibull and lognormal distributions.

Let T be the random variable for the lifetime of a power transformer, and we use the log-location-scale distribution to model T . The cumulative distribution function (cdf) of T is

$$F(t; \mu, \sigma) = \Phi \left[\frac{\log(t) - \mu}{\sigma} \right].$$

Here μ is the location parameter and σ is the scale parameter. The probability density function (pdf) of T is

$$f(t; \mu, \sigma) = \frac{1}{\sigma t} \phi \left[\frac{\log(t) - \mu}{\sigma} \right].$$

The cdf and pdf of the lognormal distribution are obtained by replacing Φ and ϕ with the standard normal cdf Φ_{nor} and pdf ϕ_{nor} , respectively. Similarly, the cdf and pdf of the Weibull distribution can be obtained by replacing Φ and ϕ with $\Phi_{\text{sev}}(z) = 1 - \exp[-\exp(z)]$ and pdf $\phi_{\text{sev}}(z) = \exp[z - \exp(z)]$.

To write down the likelihood function, we need to define some notation of the data first. Given a set of n units, the lifetime or time in service of unit i is denoted by t_i . The censoring time is denoted by c_i , and δ_i is the censoring

indicator (i.e., $\delta_i = 1$ if unit i failed and $\delta_i = 0$ if the unit was censored). The left truncation time is denoted by τ_i^L . Note that τ_i^L is the time that the life distribution of unit i was truncated on the left. The truncation indicator is v_i (i.e., $v_i = 0$ if unit i is left truncated and $v_i = 1$ if the unit is not truncated). With this notation, the likelihood function for the lifetime data of the transformers is

$$L(\boldsymbol{\theta}) = \prod_{i=1}^n [1 - F(c_i; \boldsymbol{\theta})]^{(1-\delta_i)v_i} \times \left[\frac{1 - F(c_i; \boldsymbol{\theta})}{1 - F(\tau_i^L; \boldsymbol{\theta})} \right]^{(1-\delta_i)(1-v_i)} \\ \times f(t_i; \boldsymbol{\theta})^{\delta_i v_i} \times \left[\frac{f(t_i; \boldsymbol{\theta})}{1 - F(\tau_i^L; \boldsymbol{\theta})} \right]^{\delta_i(1-v_i)},$$

where $\boldsymbol{\theta}$ is the vector that contains all the unknown parameters (i.e., the location parameters μ_i and scale parameters σ_i).

To incorporate simple covariate information, the parametric regression analysis for lifetime data is used. In a typical setting, the location parameter μ is related to the covariate \mathbf{x} through $\mu(\mathbf{x}) = \mathbf{x}'\boldsymbol{\beta}$, where $\boldsymbol{\beta}$ is the regression coefficient vector. The maximum likelihood (ML) estimate $\hat{\boldsymbol{\theta}}$ can then be obtained by maximizing the likelihood function. Numerical algorithms are usually needed.

6.2.3 Reliability Prediction

Based on the model built for the lifetime data, the prediction of the remaining life of surviving units can be obtained. The conditional distribution is important for this task. Conditional on surviving until t_i , the cdf is

$$F(t|t_i; \boldsymbol{\theta}) = \Pr(T \leq t | T > t_i) = \frac{F(t; \boldsymbol{\theta}) - F(t_i; \boldsymbol{\theta})}{1 - F(t_i; \boldsymbol{\theta})}, \quad t \geq t_i.$$

For an individual prediction, the prediction interval (PI) $[\underline{S}_i, \tilde{S}_i]$ can be used. A simple PI can be obtained by taking the lower and upper $\alpha/2$ quantiles of the estimated conditional distribution. The simple PI then needs to be calibrated to provide better coverage probability. The details of calibration using the random weighted bootstrap are available in [46].

For the prediction at the population level, the cumulative number of failures at a future time point is used. For a surviving unit i , the conditional probability of failure between age t_i and a future age t_i^w can be calculated as $\rho_i = F(t_i^w | t_i, \boldsymbol{\theta})$, and it can be estimated by $\hat{\rho}_i = F(t_i^w | t_i, \hat{\boldsymbol{\theta}})$. The cumulative number of future failures is $N = \sum_{i=1}^{n^*} I_i$. Here $I_i \sim$

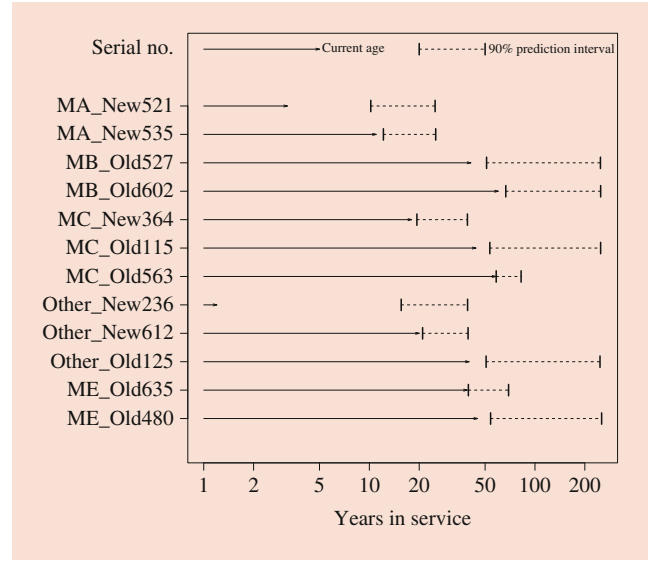


Fig. 6.2 Plot of individual PIs for the remaining life for a subset of units

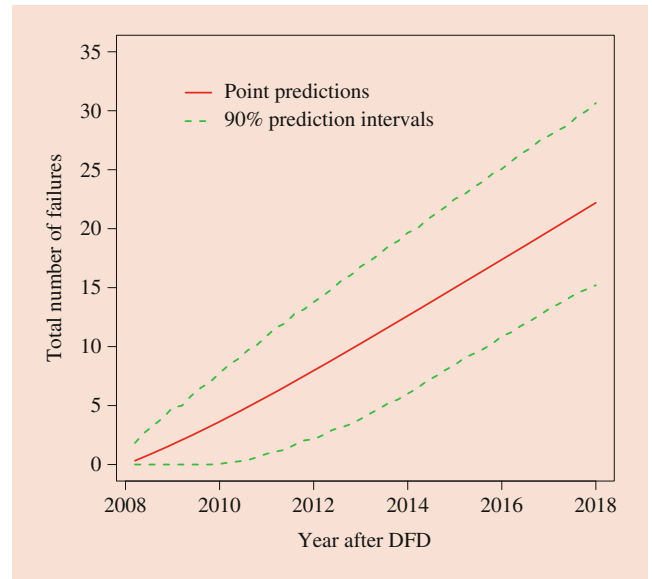


Fig. 6.3 Prediction results for the cumulative number of failures at a future time

Bernoulli(ρ_i), $i = 1, \dots, n^*$ and n^* is the number of units at risk. Based on the distribution of N , the point prediction and PIs can be obtained using the algorithms proposed in [46].

As illustrations, Fig. 6.2 shows the plot of individual PIs for the remaining life for a subset of units. Figure 6.3 visualizes the predicted cumulative number of failures over the next 10 years after DFD. One can see that the PIs are wide due to the limited covariate information in the traditional reliability data.

6.3 Time-to-Event Data Analysis

In this section, we briefly introduce the application of using dynamic covariates for time-to-event prediction as described in Hong and Meeker [10].

6.3.1 Background and Data

A general method was developed by Hong and Meeker [10] to model failure-time data with dynamic covariates. The work was motivated by the Product D2 application, which is a machine used in office/residence. Product D2 is similar to high-end copy machine where the number of pages used could be recorded dynamically. For this product, the use-rate data $R(t)$ (cycles/week) were collected weekly as a time series for those units connected to the network. This information could be downloaded automatically in addition to the failure-time data. In the Product D2 dataset, data were observed within a 70-week period, and 69 out of 1800 units failed during the study period. Figures 6.4 and 6.5 illustrate the event plot of the failure-time data and the use-rate trajectories over time for a subset of the data.

6.3.2 Model for Time-to-Event Data and Parameter Estimation

Three sets of observable random variables, the failure time, censoring indicator, and dynamic covariate, over time are considered, which are denoted by $\{T, \Delta, X(T)\}$.

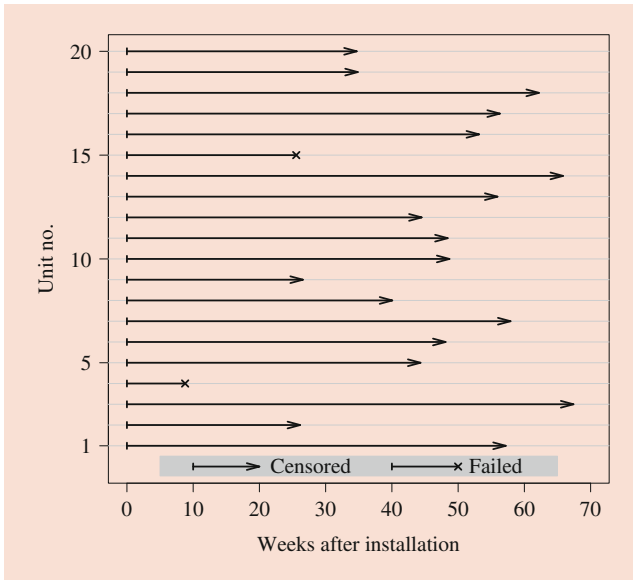


Fig. 6.4 The event plot for a subset of the Product D2 failure-time data. Figure reproduced with permission from Taylor and Francis

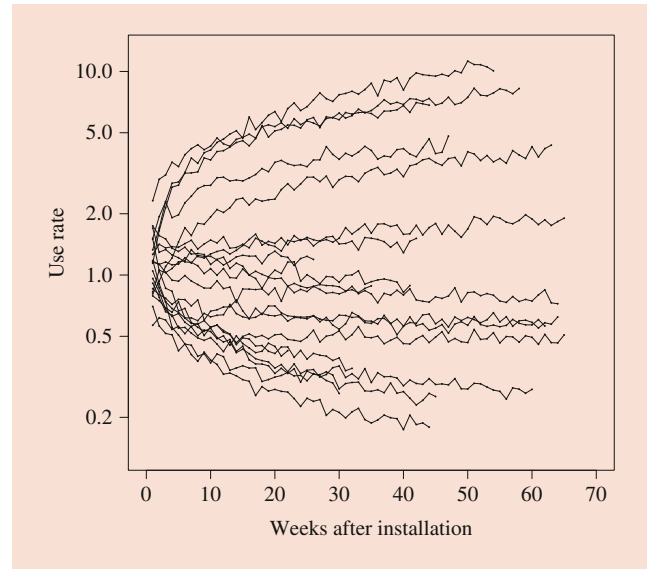


Fig. 6.5 The plot of the use-rate trajectories for the subset in Fig. 6.4. Figure reproduced with permission from Taylor and Francis

The observed data are described by $\{t_i, \delta_i, \mathbf{x}_i(t_i)\}$. Here n denotes the number of units in the dataset, t_i is the failure time or time in service, and δ_i is the observed censoring indicator (i.e., it equals to 1 if unit i fails and 0 otherwise). The $\mathbf{x}_i(t_i) = \{x_i(s) : 0 \leq s \leq t_i\}$ is the observed dynamic covariate information of unit i from the time 0 to t_i , where $x_i(s)$ is the observed covariate value at time s for unit i . Particularly for Product D2, we use $X(t) = \log[R(t)/R_0(t)]$ as the form of the covariate in the model, where $R_0(t)$ is the baseline use rate that is chosen to be a typical constant use rate.

The cumulative exposure model in [47] is used to model the failure-time data with dynamic covariate. The cumulative exposure $u(t)$ is defined as

$$u(t) = u[t; \beta, \mathbf{x}(t)] = \int_0^t \exp[\beta x(s)] ds,$$

where β represents the influence of the covariate on the exposure. When the cumulative exposure of a unit reaches a random threshold U at time T , the unit fails. This establishes a relationship between U and T as in

$$U = u(T) = \int_0^T \exp[\beta x(s)] ds. \quad (6.1)$$

Under the above model and the covariate history $\mathbf{x}(\infty)$, the cdf of the failure time T is

$$\begin{aligned} F(t; \beta, \theta_0) &= \Pr(T \leq t) = \Pr\{U \leq u[t; \beta, \mathbf{x}(t)]\} \\ &= F_0\{u[t; \beta, \mathbf{x}(t)]; \theta_0\} \end{aligned}$$

and pdf is $f(t; \beta, \boldsymbol{\theta}_0) = \exp[\beta x(t)] f_0\{u[t; \beta, \mathbf{x}(t)]; \boldsymbol{\theta}_0\}$. Here $\boldsymbol{\theta}_0$ is the parameter in the baseline cdf of the cumulative exposure threshold U , and $f_0(u; \boldsymbol{\theta}_0)$ is the pdf of U . In the Product D2 application, the baseline cumulative exposure distribution $F_0(u; \boldsymbol{\theta}_0)$ was modeled by the Weibull distribution, of which the cdf and pdf are

$$F_0(u; \boldsymbol{\theta}_0) = \Phi_{\text{sev}} \left[\frac{\log(u) - \mu_0}{\sigma_0} \right] \text{ and}$$

$$f_0(u; \boldsymbol{\theta}_0) = \frac{1}{\sigma_0 u} \phi_{\text{sev}} \left[\frac{\log(u) - \mu_0}{\sigma_0} \right].$$

In the above expression, $\boldsymbol{\theta}_0 = (\mu_0, \sigma_0)'$, where μ_0 and σ_0 are the location and scale parameters. Also, $\Phi_{\text{sev}}(z) = 1 - \exp[-\exp(z)]$, and $\phi_{\text{sev}}(z) = \exp[z - \exp(z)]$. Lognormal and other log-location-scale distributions can also be used if they are considered appropriate for certain applications.

6.3.3 Model for Covariates

To model the covariate process, we use the linear mixed-effect model. In particular, $X(t)$ is modeled as

$$X_i(t_{ij}) = \eta + Z_i(t_{ij})\mathbf{w}_i + \varepsilon_{ij}. \quad (6.2)$$

In model (6.2), η is the constant mean, and the term $Z_i(t_{ij})\mathbf{w}_i$ is used to model variation at individual level. Here $Z_i(t_{ij}) = [1, \log(t_{ij})]$, and \mathbf{w}_i is the vector of random effects of the initial covariate at time 0 and the changing rate for unit i . It is assumed that $\mathbf{w}_i = (w_{0i}, w_{1i})' \sim N(\mathbf{0}, \boldsymbol{\Sigma}_w)$ with the covariance matrix

$$\boldsymbol{\Sigma}_w = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix},$$

and $\varepsilon_{ij} \sim N(0, \sigma^2)$ is the error term.

The parameter estimation is done separately using the ML method in two parts since parameters in the failure-time model $\boldsymbol{\theta}_T = (\boldsymbol{\theta}'_0, \beta)'$ and covariate process model $\boldsymbol{\theta}_X = (\eta, \boldsymbol{\Sigma}_w, \sigma^2)$ are separable. The joint likelihood for $\boldsymbol{\theta}_T$ and $\boldsymbol{\theta}_X$ is

$$L(\boldsymbol{\theta}_T, \boldsymbol{\theta}_X) = L(\boldsymbol{\theta}_T) \times L(\boldsymbol{\theta}_X). \quad (6.3)$$

The first component of (6.3) is the likelihood function of the failure-time data, which is

$$L(\boldsymbol{\theta}_T) = \prod_{i=1}^n \{\exp[\beta x_i(t_i)] f_0(u[t_i; \beta, \mathbf{x}_i(t_i)]; \boldsymbol{\theta}_0)\}^{\delta_i} \\ \times \{1 - F_0(u[t_i; \beta, \mathbf{x}_i(t_i)]; \boldsymbol{\theta}_0)\}^{1-\delta_i}. \quad (6.4)$$

The second component of (6.3) is the likelihood of covariate data, which is

$$L(\boldsymbol{\theta}_X) = \prod_{i=1}^n \int_{\mathbf{w}_i} \left\{ \prod_{t_{ij} \leq t_i} f_1[x_i(t_{ij}) - \eta - Z_i(t_{ij})\mathbf{w}_i; \sigma^2] \right\} f_2(\mathbf{w}_i; \boldsymbol{\Sigma}_w) d\mathbf{w}_i. \quad (6.5)$$

In the above equation, $f_1(\cdot)$ is the pdf of a univariate normal, and $f_2(\cdot)$ is the pdf of a bivariate normal distribution.

6.3.4 Reliability Prediction

In order to predict future field failures, the distribution of the remaining life (DRL) is considered in the prediction procedure. The DRL describes the distribution of T_i for unit i given $T_i > t_i$ and $\mathbf{X}_i(t_i) = \mathbf{x}_i(t_i)$. Particularly, the probability of unit i failing within the next s time units given it has survived by time t_i is

$$\rho_i(s; \boldsymbol{\theta}) = \Pr[t_i < T_i \leq t_i + s | T_i > t_i, \mathbf{X}_i(t_i)], \quad s > 0, \quad (6.6)$$

where $\boldsymbol{\theta}$ denotes all the parameters. Then $\rho_i(s; \boldsymbol{\theta})$ can be further expressed as

$$\rho_i(s; \boldsymbol{\theta}) \\ = \mathbf{E}_{X_i(t_i, t_i+s) | X_i(t_i) = \mathbf{x}_i(t_i)} \{ \Pr[t_i < T_i \leq t_i + s | T_i > t_i, \\ \mathbf{X}_i(t_i), \mathbf{X}_i(t_i, t_i + s)] \} \quad (6.7) \\ = \frac{\mathbf{E}_{X_i(t_i, t_i+s) | X_i(t_i) = \mathbf{x}_i(t_i)} \{ F_0(u[t_i + s; \beta, \mathbf{X}_i(t_i, t_i + s)]; \boldsymbol{\theta}_0) \\ - F_0(u[t_i; \beta, \mathbf{x}_i(t_i)]; \boldsymbol{\theta}_0) \}}{1 - F_0(u[t_i; \beta, \mathbf{x}_i(t_i)]; \boldsymbol{\theta}_0)}$$

where $\mathbf{X}_i(t_1, t_2) = \{X_i(s) : t_1 < s \leq t_2\}$. Since model (6.2) is assumed for $\mathbf{X}_i(t_i, t_i + s)$ and $\mathbf{X}_i(t_i) = \mathbf{x}_i(t_i)$, the multivariate normal distribution theory can be used to obtain the conditional distribution.

The Monte Carlo simulation is used to evaluate $\rho_i(s; \hat{\boldsymbol{\theta}})$ since an analytical expression for $\rho_i(s; \boldsymbol{\theta})$ is unavailable. The following procedure is used to compute $\rho_i(s; \hat{\boldsymbol{\theta}})$:

1. Substitute $\boldsymbol{\theta}_X$ with the ML estimates $\hat{\boldsymbol{\theta}}_X$ in the distribution of $\mathbf{X}_i(t_i, t_i + s) | \mathbf{X}_i(t_i) = \mathbf{x}_i(t_i)$, and draw $\mathbf{X}_i^*(t_i, t_i + s)$ from the distribution.
2. Let $\mathbf{X}_i^*(t_i + s) = \{\mathbf{x}_i(t_i), \mathbf{X}_i^*(t_i, t_i + s)\}$ be the simulated covariate process in the time interval $(t_i, t_i + s)$.
3. Compute the DRL given $\mathbf{X}_i^*(t_i, t_i + s)$ and the ML estimates $\hat{\boldsymbol{\theta}}_T$ of $\boldsymbol{\theta}_T = (\boldsymbol{\theta}'_0, \beta)$ by

$$\rho_i^*(s; \hat{\boldsymbol{\theta}}) = \frac{F_0(u[t_i + s; \hat{\beta}, \mathbf{X}_i^*(t_i, t_i + s)]; \hat{\boldsymbol{\theta}}_0) - F_0(u[t_i; \hat{\beta}, \mathbf{x}_i(t_i)]; \hat{\boldsymbol{\theta}}_0)}{1 - F_0(u[t_i; \hat{\beta}, \mathbf{x}_i(t_i)]; \hat{\boldsymbol{\theta}}_0)}.$$

4. Repeat steps 1–3 M times and obtain $\rho_i^{*m}(s; \hat{\boldsymbol{\theta}})$, $m = 1, \dots, M$.
5. The estimate is computed by $\rho_i(s; \hat{\boldsymbol{\theta}}) = M^{-1} \sum_{m=1}^M \rho_i^{*m}(s; \hat{\boldsymbol{\theta}})$.

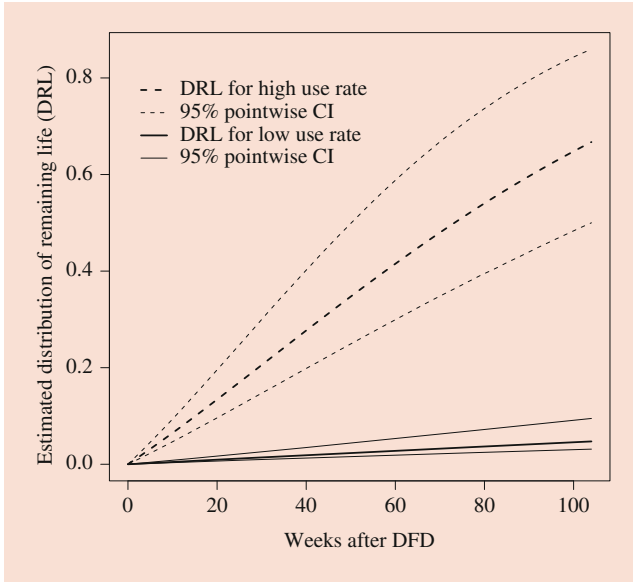


Fig. 6.6 The estimated DRLs and the 95% pointwise CIs for two representative at-risk units. Figure reproduced with permission from Taylor and Francis

The CONFIDENCE INTERVALS (CIs) for $\rho_i(s; \hat{\theta})$ can be obtained through the following procedure:

1. Draw $\hat{\theta}_T^{*b}$ and $\hat{\theta}_X^{*b}$ from $N(\hat{\theta}_T, \Sigma_{\hat{\theta}_T})$ and $N(\hat{\theta}_X, \Sigma_{\hat{\theta}_X})$, respectively.
2. Let $\hat{\theta}^* = (\hat{\theta}_T^{*b}, \hat{\theta}_X^{*b})'$ and obtain $\rho_i^{**}(s; \hat{\theta}^*)$ following the above algorithm.
3. Repeat steps 1–2 B times to obtain $\rho_i^{**b}(s) = \rho_i^{**b}(s; \hat{\theta}^{*b})$, $b = 1, \dots, B$.
4. The $100(1 - \alpha)\%$ CI is computed by $[\rho_i^{**[\alpha B/2]}(s), \rho_i^{**[(1-\alpha/2)B]}(s)]$. Here $\rho_i^{**[b]}(s)$ is the $[b]$ th ordered value of $\rho_i^{**b}(s)$, and $[\cdot]$ is the function for rounding to the nearest integer.

Figure 6.6 shows the estimated DRL for two representative units. One unit has a higher use rate which increases quickly over time ($\hat{w}_0 = 0.4061$, $\hat{w}_1 = 0.4184$), and the other has a lower use rate which increases slowly over time ($\hat{w}_0 = 0.1704$, $\hat{w}_1 = 0.0168$). The trends in the plot are consistent with our expectation that the unit with a higher use rate tends to have higher failure risk.

To assess the prediction variability, one may also want to calculate the prediction interval (PI) of an individual remaining life, denoted by $[\underline{S}_i, \tilde{S}_i]$. A $100(1 - \alpha)\%$ PI of the remaining lifetime can be obtained by using the method introduced by Lawless and Fredette [48] as in

$$\rho_i(\underline{S}_i; \hat{\theta}) = v_{\alpha/2} \quad \text{and} \quad \rho_i(\tilde{S}_i; \hat{\theta}) = v_{1-\alpha/2}. \quad (6.8)$$

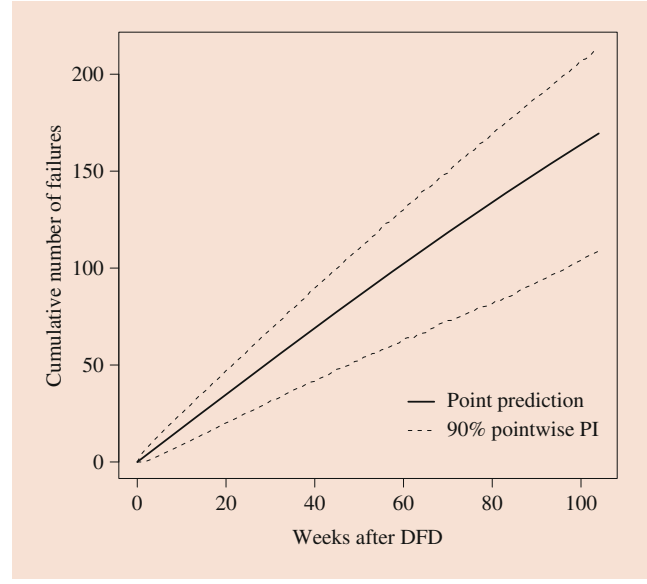


Fig. 6.7 The point predictions and the pointwise 90% PIs for the cumulative number of failures after the DFD out of the 1731 units at risk. Figure reproduced with permission from Taylor and Francis

Here v_α is the α quantile of the $\rho_i(S_i; \hat{\theta})$ distribution, and S_i represents the remaining life for unit i . The v_α can be obtained through a Monte Carlo simulation.

In real applications, obtaining the predicted cumulative number of failures is also important because this could help with the decisions for warranty cost control or the long-term production plan. Suppose $N(s) = \sum_{i \in RS} I_i(s)$ is the total failure counts at time s after DFD. The RS is the risk set at DFD in this expression, and $I_i(s)$ is the binary indicator for the occurrence of a failure at time s with $I_i(s) \sim \text{Bernoulli}[\rho_i(s; \theta)]$. Let $F_N(n_k; \theta)$, $n_k = 0, 1, \dots, n^*$ denote the cdf of $N(s)$, where n^* is the count of units in the risk set. Then $F_N(n_k; \theta)$ can be computed in its explicit form using a discrete Fourier transformation [49]. The PI for $N(s)$ can be calculated similarly as the individual predictions.

For the Product D2 application, 1731 units were remained at risk after 69 failed out of 1800 installed units by the end of the study. The point predictions and the 90% pointwise PIs of the total number of failures for 100 weeks after the DFD are shown in Fig. 6.7.

6.4 Degradation Data Analysis

In this section, we briefly introduce how to leverage the dynamic covariates for modeling degradation data as described in Hong et al. [25].

6.4.1 Background and Data

Hong et al. [25] develop a general path model utilizing shape-restricted splines with random effects to model the degradation process with dynamic covariates. This paper considers an application of modeling the photodegradation process of organic coating materials due to exposure to the time-varying ultraviolet (UV) radiation and the outdoor environmental conditions. In this work, to study the service life of a coating material, scientists at NIST placed 36 specimens in outdoor setting with varied UV spectrum and intensity, temperature, and relative humidity (RH) recorded over an approximate 5-year period. The specimens started at different time points to allow different degradation paths to be observed. For each specimen, degradation measurements were taken periodically using Fourier transform infrared (FTIR) spectroscopy. Since a particular compound or structure is tied with a peak at a certain wavelength on the FTIR spectrum, the change in the height of the peak was used to measure the decrease in the concentration of the compound. One of the compounds of interest for the NIST data was C-O stretching of aryl ether, which was measured at the wavelength 1250 cm. Figure 6.8 shows the degradation paths of nine representative specimens with varied starting times in the study. We can observe very different trajectories with the degradation rate varying over time and among different specimens as well. Figures 6.9, 6.10, and 6.11 show the dynamic covariate information on the daily UV dosage, RH, and temperature as well as the fitted smooth lines for showing the mean process of one specimen over the study period. The vertical lines are used to label time windows separated by every 6 months. We can observe both a seasonal pattern and a random oscillation

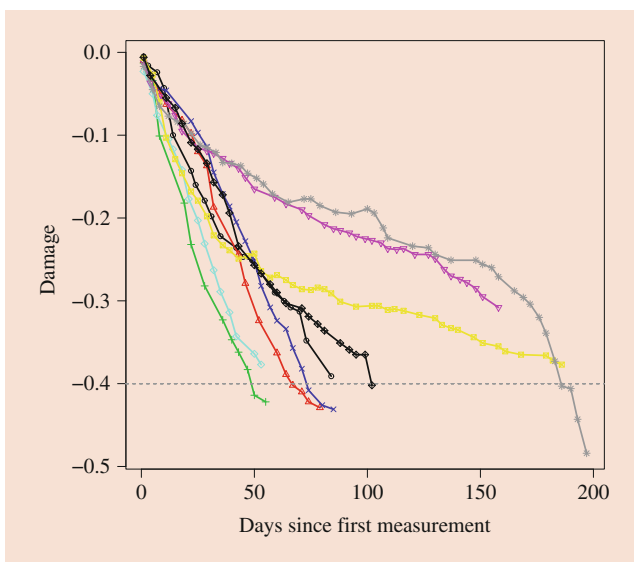


Fig. 6.8 Plot of nine representative degradation paths. *Figure reproduced with permission from Taylor and Francis*

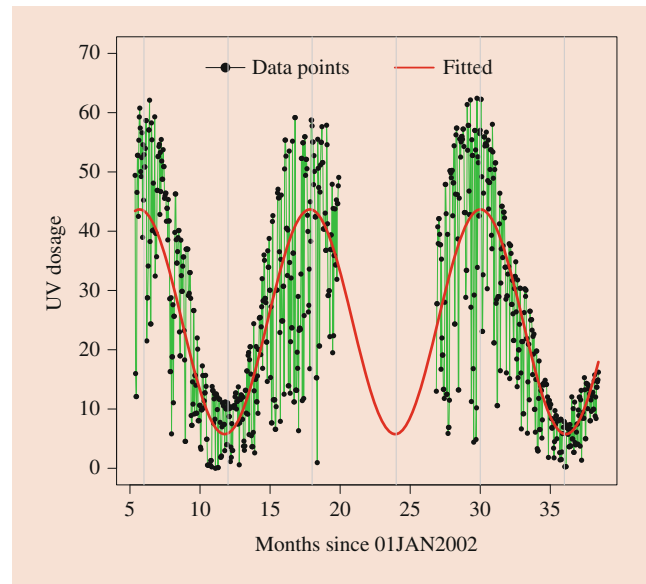


Fig. 6.9 Dynamic covariate information on the daily UV dosage for a single sample. The black dots connected by green lines show the daily values. The vertical lines show the time windows by every 6 months from January 2002. The red smooth curves are the estimated mean process. *Figure reproduced with permission from Taylor and Francis*

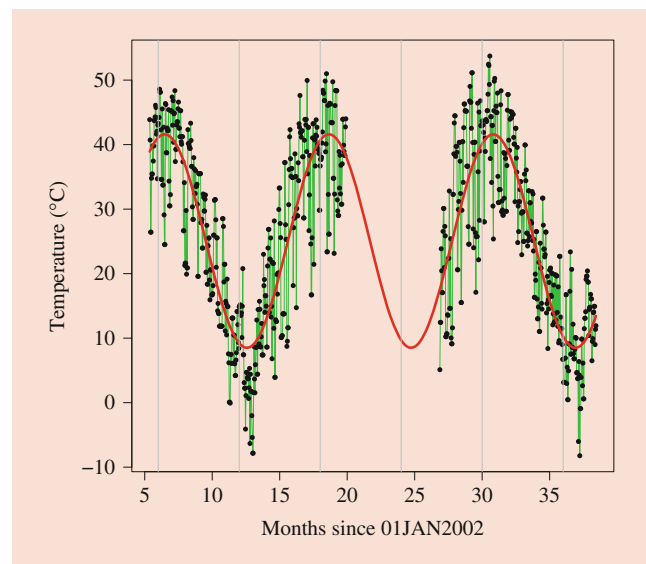


Fig. 6.10 Dynamic covariate information on the daily temperature for a single sample. The black dots connected by green lines show the daily values. The vertical lines show the time windows by every 6 months from January 2002. The red smooth curves are the estimated mean process. *Figure reproduced with permission from Taylor and Francis*

of the daily records for each individual covariate. There are stronger seasonal patterns for the UV dosage and temperature than the RH. There also appears to be a larger variation of the daily observations in the summer than in the winter, which indicates a varied degree of variability of the covariates over different time periods.

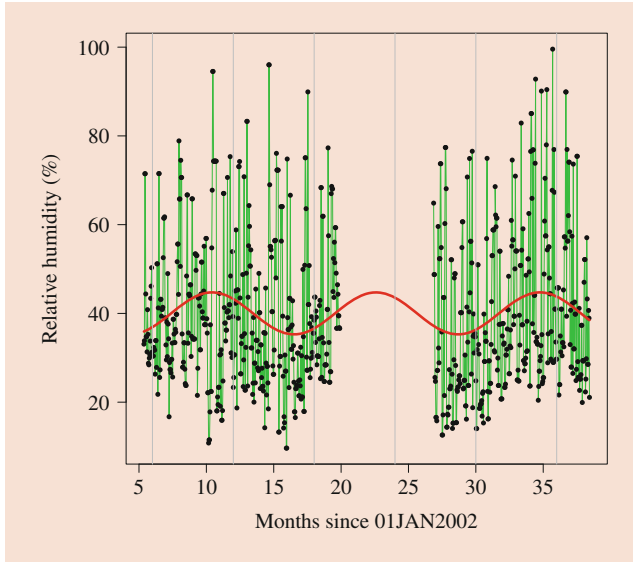


Fig. 6.11 Dynamic covariate information on the daily RH for a single sample. The black dots connected by green lines show the daily values. The vertical lines show the time windows by every 6 months from January 2002. The red smooth curves are the estimated mean process. Figure reproduced with permission from Taylor and Francis

6.4.2 Model for Degradation Paths and Parameter Estimation

The general additive model for degradation data with dynamic covariates is given in the form

$$y_i(t_{ij}) = D[t_{ij}; \mathbf{x}_i(t_{ij})] + G(t_{ij}; \mathbf{w}_i) + \varepsilon_i(t_{ij}), \quad (6.9)$$

where $y_i(t_{ij})$ for $i = 1, \dots, n, j = 1, \dots, n_i$ is the degradation measurement at time t_{ij} for unit i , $\varepsilon_i(t_{ij}) \sim N(0, \sigma_\varepsilon^2)$ denotes the measurement error, and $\mathbf{x}_i(t_{ij}) = [x_{i1}(t_{ij}), \dots, x_{ip}(t_{ij})]'$ is the vector containing the dynamic covariate information at the time t_{ij} . The actual degradation level at t_{ij} is modeled by $D[t_{ij}; \mathbf{x}_i(t_{ij})] + G(t_{ij}; \mathbf{w}_i)$ as the sum of a fixed component and a random component. The fixed component is the population common degradation path, modeled in a cumulative damage form given by

$$D[t_{ij}; \mathbf{x}_i(t_{ij})] = \beta_0 + \sum_{l=1}^p \int_0^{t_{ij}} f_l[x_{il}(u); \beta_l] du. \quad (6.10)$$

This model incorporates the dynamic covariates through the covariate-effect functions $f_l(\cdot)$ for $l = 1, \dots, p$. Here, β_0 is the initial degradation, $f_l[x_{il}(u); \beta_l]$ is the l th covariate-effect of $x_{il}(u)$ on the degradation process at time u , and $\int_0^{t_{ij}} f_l[x_{il}(u); \beta_l] du$ is the cumulative effect of x_{il} up to time t_{ij} . The random component includes the random effect terms

for modeling the unit-to-unit variation, which is specified in $G(t_{ij}; \mathbf{w}_i) = w_{0i} + w_{1i}t_{ij}$. Here, $\mathbf{w}_i = (w_{0i}, w_{1i})'$ is the vector of random effects for the initial degradation and the growth rate over time, and it is assumed to follow a bivariate normal distribution $N(\mathbf{0}, \Sigma_w)$ with the covariance matrix

$$\Sigma_w = \begin{bmatrix} \sigma_0^2 & \rho\sigma_0\sigma_1 \\ \rho\sigma_0\sigma_1 & \sigma_1^2 \end{bmatrix}.$$

Also we use $\boldsymbol{\sigma}_w = (\sigma_0, \sigma_1, \rho)'$ to denote all the distinct parameters included in Σ_w .

The ML method is used for estimating the parameters. Since the degradation measurements and the dynamic covariates are observed at discrete time points, the discrete version of the degradation path model is used for computing the likelihood by replacing $D[t_{ij}; \mathbf{x}_i(t_{ij})]$ in (6.10) by

$$D[t_{ij}; \mathbf{x}_i(t_{ij})] = \beta_0 + \sum_{l=1}^p \sum_{u_k \leq t_{ij}} f_l[x_{il}(u_k); \beta_l](u_k - u_{i,k-1}), \quad (6.11)$$

where u_k is the k th time point when the degradation and covariates are measured for unit i and $u_{i0} = 0$. Let $\boldsymbol{\theta}_D = \{\beta, \boldsymbol{\sigma}_w, \sigma_\varepsilon\}$ denote all the model parameters. Then the likelihood is

$$L(\boldsymbol{\theta}_D) = \prod_{i=1}^n \int_{\mathbf{w}_i} \left[\prod_{t_{ij} \leq t_{m_i}} \frac{1}{\sigma_\varepsilon} \phi \left\{ \frac{C[y_i(t_{ij}); \mathbf{x}_i(t_{ij}), \mathbf{w}_i]}{\sigma_\varepsilon} \right\} g_{\mathbf{w}_i}(\mathbf{w}_i; \boldsymbol{\sigma}_w) \right] d\mathbf{w}_i \quad (6.12)$$

where $C[y_i(t_{ij}); \mathbf{x}_i(t_{ij}), \mathbf{w}_i] = y_i(t_{ij}) - D[t_{ij}; \mathbf{x}_i(t_{ij})] - G(t_{ij}; \mathbf{w}_i)$, $\phi(\cdot)$ and $g_{\mathbf{w}_i}(\cdot)$ are the pdfs of a standard normal distribution and a bivariate $N(\mathbf{0}, \Sigma_w)$ distribution, respectively.

Considering there was not sufficient knowledge on what might be a sensible form for the covariate-effect functions, the paper chose to estimate the $f_l(\cdot)$ using a linear combination of spline bases. To leverage the physical understanding of the relationships between the degradation process and the covariates, the shape-restricted splines [50] were used to ensure monotonic decreasing bases (I-splines) for the UV dosage and temperature and concave bases (C-splines) for the RH. Let $B_{lq}[x_{il}(u_k)]$ for $q = 1, \dots, a_l$ denote the spline bases for the covariate x_i ; then, the covariate-effect function is modeled as

$$f_l[x_{il}(u_k); \beta_l] = \sum_{q=1}^{a_l} B_{lq}[x_{il}(u_k)] \beta_{lq},$$

where β_{lq} 's are the spline coefficients. Define $U_{lq}(t_{ij}) = \sum_{u_{ik} \leq t_{ij}} B_{lq}[x_{il}(u_{ik})](u_{ik} - u_{i,k-1})$. Then the model in (6.9) with $D[t_{ij}; \mathbf{x}_i(t_{ij})]$ given in (6.11) can be written as a linear mixed-effect model in the form of $\mathbf{y}_i = \mathbf{X}_i \boldsymbol{\beta} + \mathbf{Z}_i \mathbf{w}_i + \boldsymbol{\varepsilon}_i$, where

$$\mathbf{X}_i = \begin{bmatrix} 1 & U_{11}(t_{i1}) & \cdots & U_{1a_1}(t_{i1}) & \cdots & U_{p1}(t_{i1}) & \cdots & U_{pa_p}(t_{i1}) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 1 & U_{11}(t_{ini}) & \cdots & U_{1a_1}(t_{ini}) & \cdots & U_{p1}(t_{ini}) & \cdots & U_{pa_p}(t_{ini}) \end{bmatrix},$$

$$\mathbf{Z}_i = \begin{bmatrix} 1 & t_{i1} \\ \vdots & \vdots \\ 1 & t_{ini} \end{bmatrix},$$

and the coefficient vector $\boldsymbol{\beta} = (\boldsymbol{\beta}'_w, \boldsymbol{\beta}'_c)'$, where $\boldsymbol{\beta}_w$ and $\boldsymbol{\beta}_c$ denote the unconstrained and constrained parameters, respectively.

The following algorithm was proposed in [25] to obtain the ML estimate $\hat{\boldsymbol{\theta}}_D$ that maximizes (6.12):

1. Initialize $\boldsymbol{\sigma}_w$ and σ_ε by fitting a linear mixed-effect model with no constraints.
2. Compute $V_i = \mathbf{Z}_i \boldsymbol{\Sigma}_w \mathbf{Z}_i' + \sigma_\varepsilon^2 \mathbf{I}_i$.
3. The mixed primal-dual bases algorithm in [51] is used to estimate $\boldsymbol{\beta}$. That is to minimize $\sum_{i=1}^n (\mathbf{y}_i - \mathbf{X}_i \boldsymbol{\beta})' V_i^{-1} (\mathbf{y}_i - \mathbf{X}_i \boldsymbol{\beta})$ subject to $\boldsymbol{\beta}_c \geq \mathbf{0}$.
4. Fit a linear mixed-effect model $r_i = \mathbf{Z}_i \mathbf{w}_i + \boldsymbol{\varepsilon}_i$ with $r_i = \mathbf{y}_i - \mathbf{X}_i \boldsymbol{\beta}$ to get the updated estimates of $\boldsymbol{\sigma}_w$ and σ_ε .
5. Repeat steps 2 to 4 until the estimated parameters converge.

With the shape-restricted splines, the ML estimates of some parameters might locate on the boundary of the parameter space. In this case, the bootstrap method is useful for assessing the variability and making inferences about the parameters. An adjusted bootstrap procedure by Carpenter et al. [52] was applied to resample the residuals and the estimated random effects for constructing bootstrap resamples of the original data to avoid underestimating variability and producing too narrow CIs. Then the bias-corrected bootstrap CIs were constructed based on obtaining the ML estimates of model parameters using the abovementioned algorithm for a large number of bootstrap samples.

6.4.3 Model for Covariates

To predict the degradation process and reliability, it is necessary to model the dynamic covariate process through a parametric model. Hong et al. [25] propose the following model:

$$X_j(t) = \text{Tr}_j(t) + S_j(t) + \xi_j(t),$$

where $\text{Tr}_j(t)$ models the long-term trend of the covariate process for the j th covariate, $S_j(t)$ captures the seasonal

pattern, and $\xi_j(t)$ depicts the random error which is modeled by a stationary process. For the NIST outdoor weathering data, there was no significant long-term trend observed, and hence $\text{Tr}_j(t) = \mu_j$ for $j = 1, 2, 3$. However, the seasonal pattern was quite prominent, and there were seasonal effects observed for both the mean and variance of the process. So two sine functions were included in both the seasonal and error terms (except for RH which shows no seasonal effect assumed for the variation of the process from Figs. 6.8, 6.9, 6.10, and 6.11) in the following form:

$$\begin{bmatrix} S_1(t) \\ S_2(t) \\ S_3(t) \end{bmatrix} = \begin{bmatrix} \kappa_1 \sin \left[\frac{2\pi}{365} (t - \eta_1) \right] \\ \kappa_2 \sin \left[\frac{2\pi}{365} (t - \eta_2) \right] \\ \kappa_3 \sin \left[\frac{2\pi}{365} (t - \eta_3) \right] \end{bmatrix},$$

$$\begin{bmatrix} \xi_1(t) \\ \xi_2(t) \\ \xi_3(t) \end{bmatrix} = \begin{bmatrix} (1 + \nu_1 \{1 + \sin \left[\frac{2\pi}{365} (t - \zeta_1) \right]\}) \varepsilon_1(t) \\ (1 + \nu_2 \{1 + \sin \left[\frac{2\pi}{365} (t - \zeta_2) \right]\}) \varepsilon_2(t) \\ \varepsilon_3(t) \end{bmatrix}. \quad (6.13)$$

To capture the autocorrelation within and among the covariate processes, a lag-2 VAR model [i.e., $\text{Var}(2)$] was used, where the error term was modeled by

$$\begin{bmatrix} \varepsilon_1(t) \\ \varepsilon_2(t) \\ \varepsilon_3(t) \end{bmatrix} = Q_1 \begin{bmatrix} \varepsilon_1(t-1) \\ \varepsilon_2(t-1) \\ \varepsilon_3(t-1) \end{bmatrix} + Q_2 \begin{bmatrix} \varepsilon_1(t-2) \\ \varepsilon_2(t-2) \\ \varepsilon_3(t-2) \end{bmatrix} + \begin{bmatrix} e_1(t) \\ e_2(t) \\ e_3(t) \end{bmatrix}. \quad (6.14)$$

In the above equation, Q_1 and Q_2 are regression coefficient matrices, and $[e_1(t), e_2(t), e_3(t)]' \sim \mathbf{N}(\mathbf{0}, \boldsymbol{\Sigma}_e)$ are multivariate normal random errors that do not change over time.

The parameters in models (6.13) and (6.14) are estimated in two steps. First, the ML estimates of the seasonal effects in the process mean and variance structures are obtained by ignoring the autocorrelation in the error terms. Then the VAR model is fitted to the residuals calculated from the first step using the multivariate least squares approach [53]. The bootstrap method is used for obtaining the CIs of the parameters in the dynamic covariate process.

6.4.4 Reliability Prediction

To predict the failure time and reliability, let \mathcal{D}_f denote the threshold for a soft failure. For any $\mathbf{X}(\infty) = \mathbf{x}(\infty)$ and \mathbf{w} , the degradation path is fixed and the failure time can be determined by

$$t_{\mathcal{D}} = \min\{t : D[t; \mathbf{x}(\infty)] + G(t; \mathbf{w}) = \mathcal{D}_f\}. \quad (6.15)$$

However, for a random unit, the covariate process $X(\infty)$ and w are random. Hence, the cdf of the failure time, $T = T[\mathcal{D}_f, X(\infty), w]$, can be defined as

$$F(t; \theta) = \mathbf{E}_{X(\infty)} \mathbf{E}_w \Pr \{T[\mathcal{D}_f, X(\infty), w] \leq t\}, \quad t > 0, \tag{6.16}$$

with $\theta = \{\theta_D, \theta_X\}$ denoting all the unknown parameters. There is usually no closed form expression of $F(t; \theta)$. Hence, the cdf at any estimated $\hat{\theta}$ is estimated through Monte Carlo simulation outlined in the following steps [25]:

1. Simulate the covariate process based on the estimated parameter $\hat{\theta}_X$.
2. Simulate the random effects w from $N(\mathbf{0}, \Sigma_w)$ with the estimated parameter $\hat{\theta}_D$.
3. Compute $D[t; X(\infty)] + G(t; w)$ based on the simulated covariate process and random effects.
4. For the degradation path in step 3, determine the failure time $t_{\mathcal{D}}$ by Eq. (6.15).
5. Repeat steps 1 to 4 for M times to obtain the simulated failure times $t_{\mathcal{D}}^m, m = 1, \dots, M$. Then $F(t; \hat{\theta})$ is estimated by the empirical cdf, $F(t; \hat{\theta}) = M^{-1} \sum_{m=1}^M \mathbb{1}_{(t_{\mathcal{D}}^m \leq t)}$.

By using the bootstrap approach, the point estimates and the CIs of $F(t; \theta)$ can be calculated using the sample mean and quantiles of the bootstrap estimates of $F(t; \hat{\theta})$ based on a large number of bootstrap estimates $\hat{\theta}$. By using $\mathcal{D}_f = -0.4, M = 200$ Monte Carlo simulations, and 10000 bootstrap samples, Fig. 6.12 shows the predicted $F(t; \theta)$ and its 95% pointwise CIs for the NIST coating degradation data. We can see that for

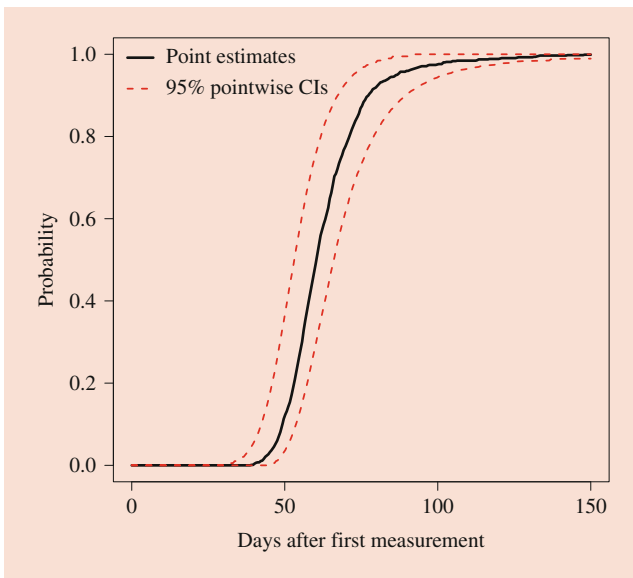


Fig. 6.12 The estimated cdf and corresponding 95% pointwise CIs for a population of units with random starting time between 161 and 190 days. Figure reproduced with permission from Taylor and Francis

a population of units with random starting time between 161 and 190 days, a majority of the population will fail between 50 and 150 days in service.

6.5 Recurrent Event Data Analysis

In this section, we briefly introduce the multilevel trend-renewal process (MTRP) model and its application on the Vehicle B data as described in Xu et al. [36].

6.5.1 Background and Data

Xu et al. [36] consider the modeling and analysis of the Vehicle B data, which consist of recurrent event data from a batch of industrial systems. Vehicle B is a two-level repairable system. During its life span, Vehicle B may experience events at subsystem level (e.g., engine failures) and/or events at component level (e.g., oil pump failures). In the field data, we have $n = 203$ units from a 110-month observation period. There are 219 component events and 44 subsystem events observed during the study period. Figure 6.13 shows the event plot for ten randomly selected units. We also have the cumulative usage information available for each unit, which is a dynamic covariate. The cumulative usage information is shown in Fig. 6.14. The goal is to make a prediction for the cumulative number of component event occurrences at a future time.

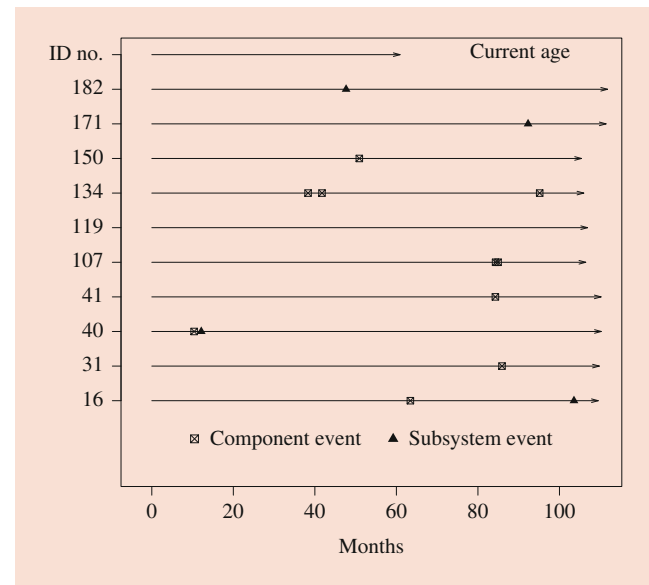


Fig. 6.13 The recurrent event processes for ten randomly selected units in the Vehicle B fleet. Figure reproduced with permission from Taylor and Francis

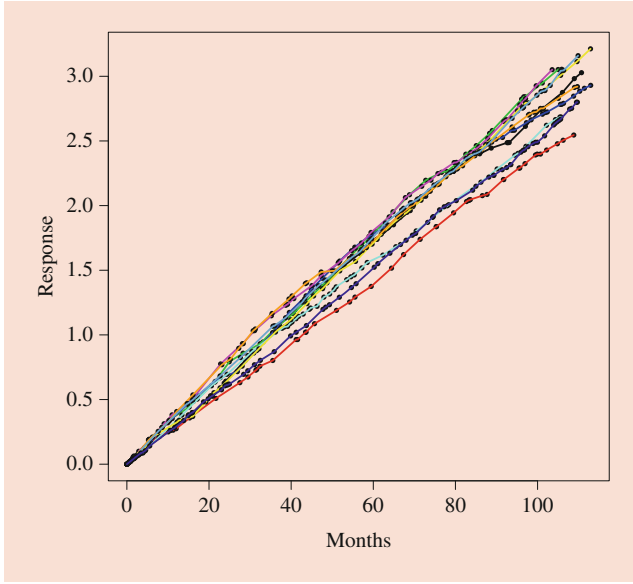


Fig. 6.14 The cumulative usage processes for ten randomly selected units in the Vehicle B fleet. Figure reproduced with permission from Taylor and Francis

We need some notation to introduce the MTRP model. Suppose there are n units under observation from time 0 to τ_i . Let $X_i(t)$ be the time-dependent covariate at time t for system i . Let $N_{is}(t)$ be the number of subsystem events, and let $N_{ic}(t)$ be the number of component events up to time t . The total number of replacement events is $N_i(t) = N_{is}(t) + N_{ic}(t)$. The subsystem event time is sorted as $0 < t_{i1}^s < \dots < t_{i,N_{is}(\tau_i)}^s < \tau_i$. The component event time is sorted as $0 < t_{i1}^c < \dots < t_{i,N_{ic}(\tau_i)}^c < \tau_i$. Let $0 < t_{i1} < \dots < t_{i,N_i(\tau_i)} < \tau_i$ be the replacement event times, regardless of the types.

6.5.2 The MTRP Model and Parameter Estimation

For a two-level repairable system, Xu et al. [36] propose the following MTRP model to describe events occurred at component level. In particular, the intensity function is

$$\lambda_i^c(t|\mathcal{F}_{i,t^-}; \theta^c) = h^c \left\{ \Lambda_i^s(t|\mathcal{F}_{i,t^-}^s) - \Lambda_i^s \left[t_{i,N_i(t^-)} | \mathcal{F}_{i,t^-}^s \right]; \theta^c \right\} \times \lambda_i^s(t|\mathcal{F}_{i,t^-}^s; \theta^c). \quad (6.17)$$

Here \mathcal{F}_{i,t^-}^s denotes the historical information. In this multi-level model framework, the effect of subsystem events on the

component event process is modeled by $\lambda_i^s(t|\mathcal{F}_{i,t^-}^s; \theta^c)$, which takes the form

$$\lambda_i^s(t|\mathcal{F}_{i,t^-}^s; \theta^c) = h^s \{ \Lambda_i(t) - \Lambda_i[t_{i,N_{is}(t^-)}^s]; \theta^c \} \lambda_i(t; \theta^c). \quad (6.18)$$

Here, θ^c denotes the vector of unknown parameters. The cumulative event intensity functions can be obtained as $\Lambda_i(t) = \int_0^t \lambda_i(u; \theta^c) du$, and $\Lambda_i^s(t|\mathcal{F}_{i,t^-}^s) = \int_0^t \lambda_i^s(u|\mathcal{F}_{i,t^-}^s; \theta^c) du$. The baseline function $\lambda_i(t; \theta^c)$ models the intensity of the component process when there is no event adjustment, and the function $h^s(\cdot)$ is used to model the adjustment for the effect of events from the subsystem. The renewal distribution function $F^c(\cdot)$ is used to describe the distribution of gap times under the transformed scale. The model in (6.18) can be extended to incorporate dynamic covariates and random effects.

To model the dynamic covariates, the intensity function can be extended as

$$\lambda_i(t; \theta^c) = \lambda_b(t) \exp\{\gamma g[X_i(t)]\}, \quad (6.19)$$

where $\lambda_b(t)$ denotes the intensity trend function under the baseline and γ is the regression coefficient. In the Vehicle B application, we use $g[X_i(t)] = \log[X_i(t)]$. To incorporate random effects, the intensity function can be further extended as

$$\lambda_i(t; \theta^c) = \lambda_b(t) \exp\{\gamma \log[X_i(t)] + w_i\}. \quad (6.20)$$

Here w_i 's are independent and identically distributed with $N(0, \sigma_r^2)$. The MTRP with random effects is referred to as HMTRP(F^c, F^s, λ_i), in which the HMTRP stands for heterogenous MTRP.

To estimate the model parameters, one needs to construct the likelihood function. The component event data can be denoted as $\{t_{ij}, \delta_{ij}^c\}$ with t_{ij} being the event time and δ_{ij}^c being the component-event indicator. The event history is denoted as $\mathcal{F} = \{N_{ic}(u), N_{is}(u), X_i(u) : 0 < u \leq \tau_i, i = 1, \dots, n\}$. The likelihood function is given by

$$L(\theta^c) = \prod_{i=1}^n \prod_{j=1}^{N_i(\tau_i)+1} \left(\left\{ f^c[\Lambda_i^s(t_{ij}|\mathcal{F}_{i,t_{ij}^-}^s) - \Lambda_i^s(t_{i,j-1}|\mathcal{F}_{i,t_{i,j-1}^-}^s)] \lambda_i^s(t_{ij}|\mathcal{F}_{i,t_{ij}^-}^s; \theta^c) \right\}^{\delta_{ij}^c} \times \left\{ S^c[\Lambda_i^s(t_{ij}|\mathcal{F}_{i,t_{ij}^-}^s) - \Lambda_i^s(t_{i,j-1}|\mathcal{F}_{i,t_{i,j-1}^-}^s)] \right\}^{1-\delta_{ij}^c} \right). \quad (6.21)$$

Xu et al. [36] use Bayesian methods with diffuse priors to estimate the model parameters. The Metropolis-within-Gibbs algorithm is used to obtain the posterior distributions, and then the inference can be carried out using the Markov chain Monte Carlo (MCMC) samples from the posterior distributions.

6.5.3 Prediction for Component Events

To make predictions for component events, let θ denote the vector of all the parameters and $X_i(t_1, t_2) = \{X_i(t); t_1 < t \leq t_2\}$ is the covariate information between t_1 and t_2 . The prediction for the counts of component events at time t^* is

$$\begin{aligned}
 N_c(t^*; \theta) &= \sum_{i=1}^n N_{ic}(t^*; \theta) \\
 &= \sum_{i=1}^n \mathbf{E}_{X_i(\tau_i, \tau_i+t^*)|X(\tau_i)} \mathbf{E}_{w_i} \{N_{ic}[t^*, X_i(\tau_i, \tau_i + t^*), \\
 &\qquad\qquad\qquad w_i; \theta]\}. \quad (6.22)
 \end{aligned}$$

Here $N_{ic}(t^*; \theta)$ denotes the prediction for unit i . Because there is no closed form expression for (6.22), the Monte Carlo simulation approach is used.

By fitting the MTRP model to the Vehicle B data using Bayesian estimation, one needs to specify the prior distributions for the unknown parameters. The Weibull distribution was used as renewal functions for F^c and F^s . To check the performance of the predictions, the last 15 months of the Vehicle B data were held back, and only the first 95 months of data were used to estimate the MTRP model. Then we generate predictions for the last 15 months. Figure 6.15 shows the predicted cumulative number of component events

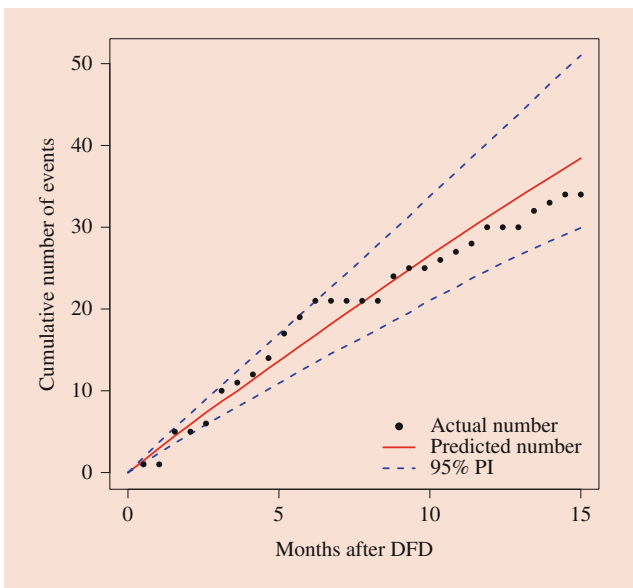


Fig. 6.15 Back test based on an early subset of the data. The plot shows the predicted cumulative number of component events for Vehicle B for the last 15 months based on the earlier 95 months of data. *Figure reproduced with permission from Taylor and Francis*

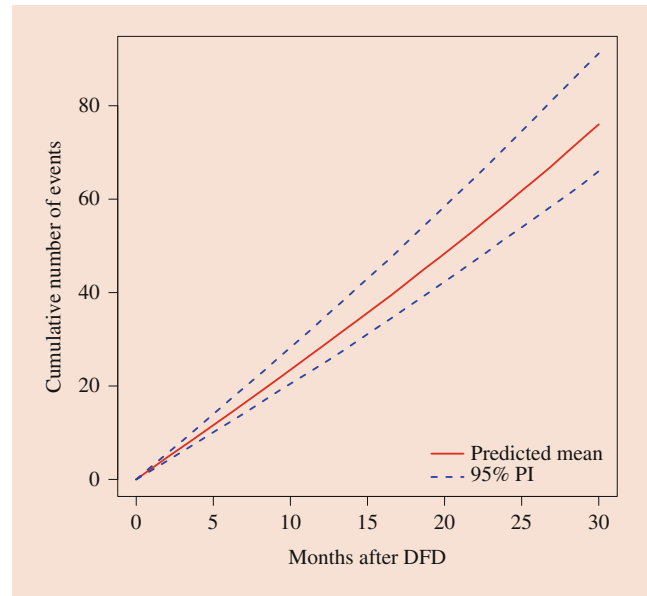


Fig. 6.16 Prediction of future events. The plot shows the predicted cumulative number of component events for Vehicle B in the future 30 months based on all observed data. *Figure reproduced with permission from Taylor and Francis*

for the last 15 months based on the earlier data. One can see that the actual observed cumulative numbers of component events are closely located around the predicted values and also well bounded within the pointwise PIs. Figure 6.16 shows the predicted future events given all the observed data for the next 30 months, which indicates that the total number of component events is expected to range between 62 and 90 with a 95% confidence level.

6.6 Sequential Test Planning of Accelerated Life Tests

In this section, we briefly introduce the sequential Bayesian design (SBD) for fatigue test experiments described in Lee et al. [43].

6.6.1 Background and Historical Data

A sequential Bayesian test planning strategy for the accelerated life tests was proposed by Lee et al. [43]. The fatigue test for glass fiber, a composite material, is considered to illustrate the sequential design strategy. In the test, a tensile/compressive stress s (positive/negative value) is applied to the test unit, and the material life is observed under that stress. In this work, 14 observations of E-glass are made including

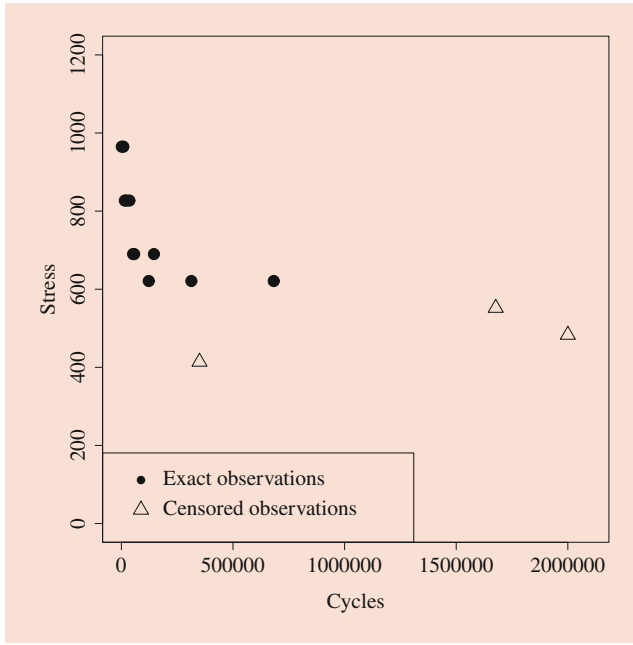


Fig. 6.17 The stress-life plot of historical data from a fatigue testing of the glass fiber. Figure reproduced with permission from Taylor and Francis

11 failed and three right censored units. Historical data of the observations are shown in Fig. 6.17. Several other important factors in the test are set as follows. Let $R = \sigma_m / \sigma_M$ denote the stress ratio, where σ_m is the minimum stress and σ_M is the maximum stress. The range of R can reveal different test types, and it is set at $R = 0.1$ for a tension-tension loading test in this application. The ultimate stress σ_{ult} , where the material breaks at the first cycle, is set to be 1339.67 MPa. The frequency of the cyclic stress testing (f) is set at 2 Hz, and the angle (α) between the testing direction and material is set at 0.

6.6.2 Lifetime Model

Consider using a log-location-scale distribution to model the cycles to failure, T . The cdf and pdf are given as

$$F(t; \theta) = \Phi \left[\frac{\log(t) - \mu}{\nu} \right] \quad \text{and}$$

$$f(t; \theta) = \frac{1}{\nu t} \phi \left[\frac{\log(t) - \mu}{\nu} \right],$$

where $\Phi(\cdot)$ and $\phi(\cdot)$ are the standard cdf and pdf, respectively. The lognormal and Weibull distributions are the common choices. In the ALT modeling, we assume a constant scale parameter ν and the location parameter is $\mu = \mu_{\beta}(x)$, where x is the stress level and β is the vector of unknown

parameters. The following nonlinear model for composite materials proposed in [54] is used to describe $\mu = \mu_{\beta}(x)$ in the form of

$$\mu_{\beta}(x) = \frac{1}{B} \log \left\{ \left(\frac{B}{A} \right) h^B \left(\frac{\sigma_{ult}}{x} - 1 \right) \left(\frac{\sigma_{ult}}{x} \right)^{\gamma(\alpha)-1} [1 - \psi(R)]^{-\gamma(\alpha)} + 1 \right\}. \quad (6.23)$$

In the above model, A and B are effects from environment and material, and $\beta = (A, B)'$. The function $\psi(R) = 1/R$ if $R \geq 1$ and $\psi(R) = R$ if $-\infty < R < 1$, and $\gamma(\alpha) = 1.6 - \psi |\sin(\alpha)|$. Then $\theta = (\beta', \nu)'$ denotes the unknown parameters in the ALT modeling.

The lower quantile of the cycles-to-failure distribution is of interest as it contains material life information. The log of the p th quantile is

$$\log(\xi_{p,u}) = \mu_{\beta}(u) + z_p \nu, \quad (6.24)$$

where $\xi_{p,u}$ is the p th quantile at the use condition u and z_p is the p th quantile of the standard distribution. Our goal is to propose test planning under multiple use conditions to approximate the real scenarios. The use stress profile consists of a set of use levels, $\{u_1, \dots, u_K\}$, with weights $\{\omega_1, \dots, \omega_K\}$ and $\sum_{k=1}^K \omega_k = 1$.

Let (x_i, t_i, δ_i) denote the data for the i th testing unit, where x_i is the stress level of the accelerating factor and t_i is the observed cycles to failure (or censoring cycles). Let δ_i be a censoring indicator where $\delta_i = 1$ if the observation is censored and $\delta_i = 0$ if the observation fails. Then, the log-likelihood function is given by

$$l(\theta | \mathbf{x}_n, \mathbf{t}_n, \boldsymbol{\delta}_n) = \sum_{i=1}^n (1 - \delta_i) [\log \phi(z_i) - \log(t_i) - \log(\nu)] + \delta_i \log [1 - \Phi(z_i)], \quad (6.25)$$

where $z_i = [\log(t_i) - \mu_{\beta}(x_i)] / \nu$. Let $\hat{\theta}$ be the ML estimate of θ and let $\log(\hat{\xi}_{p,u})$ be the ML estimate of the logarithm of the p th quantile at the use level u , obtained by substituting β and ν by $\hat{\beta}$ and $\hat{\nu}$ in (6.24). Given the use level u , the asymptotic variance of $\log(\hat{\xi}_{p,u})$ is

$$\text{Avar}[\log(\hat{\xi}_{p,u})] = \mathbf{c}' \Sigma_{\theta}(\mathbf{x}_n) \mathbf{c},$$

where $\mathbf{c} = [\partial \mu_{\beta}(u) / \partial A, \partial \mu_{\beta}(u) / \partial B, z_p]'$, $\Sigma_{\theta}(\mathbf{x}_n) = I_n^{-1}(\theta)$, and $I_n(\theta)$ is the Fisher information matrix based on n observed data. The details for calculating $I_n(\theta)$ can be found in [43]. A weighted version of asymptotic variance can be expressed as

$$\sum_{k=1}^K \omega_k \text{Avar}[\log(\hat{\xi}_{p,u_k})]. \quad (6.26)$$

Given $\{(u_k, \omega_k)\}_{k=1}^K$, the weighted asymptotic variance only depends on the observed testing levels x_i , where $i = 1, \dots, n$.

Therefore, the optimum design should determine x_1, \dots, x_n to minimize the weighted asymptotic variance in (6.26).

6.6.3 Test Plan Development

To obtain an accurate prediction from an efficient ALT, the optimum test planning can be determined by minimizing the asymptotic variance in (6.26). In the literature, when determining an optimal test plan, it often requires pre-specifying the values of parameters (known as the planning values). The optimal design based on some chosen planning values of parameters is known as the local c -optimality design. However, the planning values are not precisely known a priori for many experiments in practice. Hence, the SBD is useful for improving our understanding of the unknown parameters as more data become available during the experiment, when there is little knowledge or historical data available.

Before the test planning, the stress levels are often standardized to be between 0 and 1, denoted by $q_i = x_i / \sigma_{ult}$. In practice, a range of testing levels, $[q_L, q_U]$, is often determined at the very early stage of the test planning, where q_L is the lower bound and q_U is the upper bound. To design an efficient ALT via the sequential planning, the objective function based on (6.26) is chosen as

$$\varphi(q_{\text{new}}) = \int_{\Theta} \left[\sum_{k=1}^K \omega_k \mathbf{c}_k' \Sigma_{\theta}(q_{\text{new}}) \mathbf{c}_k \right] \pi(\boldsymbol{\theta} | \mathbf{q}_n, \mathbf{t}_n, \boldsymbol{\delta}_n) d\boldsymbol{\theta}, \quad (6.27)$$

where $\Sigma_{\theta}(q_{\text{new}}) = [I_n(\boldsymbol{\theta}, \mathbf{q}_n) + I_1(\boldsymbol{\theta}, q_{\text{new}})]^{-1}$, $\mathbf{q}_{\text{new}} = (q'_n, q_{\text{new}})'$, $\mathbf{q}_n = (q_1, \dots, q_n)'$, and $\pi(\boldsymbol{\theta} | \mathbf{q}_n, \mathbf{t}_n, \boldsymbol{\delta}_n)$ is the posterior distribution of $\boldsymbol{\theta}$. Specifically,

$$\pi(\boldsymbol{\theta} | \mathbf{q}_n, \mathbf{t}_n, \boldsymbol{\delta}_n) \propto f(\mathbf{t}_n | \boldsymbol{\theta}, \mathbf{x}_n, \boldsymbol{\delta}_n) \pi(\boldsymbol{\theta}),$$

where $f(\mathbf{t}_n | \boldsymbol{\theta}, \mathbf{x}_n, \boldsymbol{\delta}_n)$ is the joint pdf of the historical data and $\pi(\boldsymbol{\theta})$ is the prior distribution of $\boldsymbol{\theta}$. Then, the optimum $(n + 1)$ th design point is determined by

$$q_{n+1}^* = \arg \min_{q_{\text{new}} \in [q_L, q_U]} \varphi(q_{\text{new}}). \quad (6.28)$$

The procedure of the sequential Bayesian design is summarized as follows:

1. *Specify prior distributions of model parameters.* Specify prior distributions of A and B as $A \sim N(\mu_A, \sigma_A^2)$ and $B \sim N(\mu_B, \sigma_B^2)$, where μ_A , σ_A^2 , μ_B , and σ_B^2 are the parameters of the normal distributions and set to be known constants.

Let $v^2 \sim \text{Inverse Gamma}(\kappa, \gamma)$, where κ and γ can be known from the historical data or experience.

2. *Evaluate the asymptotic variance.* Use the technique of MCMC to approximate (6.27). The details of the related algorithms can be found in [43].
3. *Determine the optimum next testing point q_{n+1}^* .* Given a candidate set of design points, their corresponding values of the objective function in (6.27) can be evaluated in step 2. Then, one can determine the optimum next design point, which offers the smallest value of the asymptotic variance.
4. *Obtain the failure data at the level q_{n+1}^* .* Under the stress level q_{n+1}^* , conduct the experiment and obtain the failure information (t_{n+1}, δ_{n+1}) .
5. *Repeat steps 2 to 4 till the desired number of testing units is obtained.* Add the new lifetime data, $(q_{n+1}^*, t_{n+1}, \delta_{n+1})$, to the historical dataset, and repeat steps 2 to 4 till the desired number of new design points is obtained.

6.6.4 Illustration of Test Plans

For the original data, we can fit the lognormal distribution, and the corresponding ML estimates are $\boldsymbol{\theta}_0 = \hat{\boldsymbol{\theta}} = (0.0157, 0.3188, 0.7259)'$. Before the testing planning, the setup for the sequential Bayesian design is as follows:

1. *Prior information:* Let A and B be from the normal distributions, where $A \sim N(0.08, 0.0008)$ and $B \sim N(1, 0.0833)$. The prior distribution for v^2 is Inverse Gamma(4.5, 3).
2. *Historical data:* In practical implementation, the sample size at the beginning of testing is limited. Hence, three failed observations at stress levels $\mathbf{x}_3 = (621, 690, 965)'$ from Fig. 6.18 were chosen as the historical dataset.
3. *Total size of design points:* Let the sample size of the new design points be 12.
4. *Design candidate:* The standardized levels of historical data are 0.46, 0.52, and 0.72, and the candidate points are from $q_L = 0.35$ to $q_U = 0.75$ with a 5% incremental step.

For the illustrative purpose, we assume the true values of parameters to be $\boldsymbol{\theta}_0$. When an optimum design point is determined, the new observation is generated from the lognormal distribution with parameter $\boldsymbol{\theta}_0$ and the censoring time at 2×10^6 cycles. Repeat steps 2 to 4 in Sect. 6.6.3 till 12 testing locations are obtained. Then, the results of four simulation trials are shown in Fig. 6.19. It consistently shows that only two stress levels at 0.35 and 0.75 are selected, and 8 and 4 units are allocated to the levels 0.35 and 0.75, respectively. Figure 6.20 shows that the resulting asymptotic variance decreases as the size of sequential runs increases for the simulation trials.

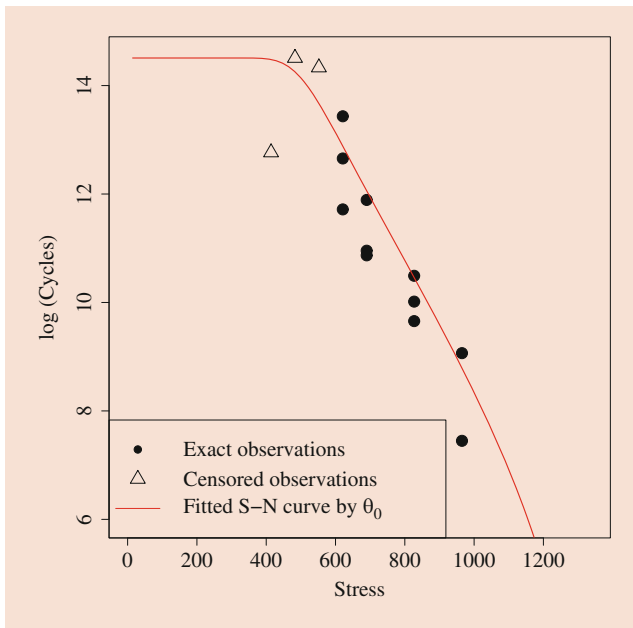


Fig. 6.18 The fitted stress-life relationship from a fatigue testing data of the glass fiber. *Figure reproduced with permission from Taylor and Francis*

Using the same historical data, the developed SBD is also compared with the local c -optimality design. For the local c -optimality design, the estimated values of parameters from historical data are usually used as the planning values of the parameters. With only three observations available from the historical data, the ML estimates are $\hat{\theta}_1 = (0.0005, 0.7429, 0.1658)'$. The resulting local c -optimality design chooses 11 and 1 unit at the testing levels at 0.65 and 0.75, respectively. Now, we compare the performance on the value of the asymptotic variance based on the ML estimates of the final dataset including the 12 new testing observations and three historical observations. With 100 simulations, the average asymptotic variances for the SBD and the local c -optimality designs are 0.6048 and 4.0337, respectively. It shows that the SBD is more efficient than the traditional local c -optimality design when there is too little historical data available to provide accurate estimates of the model parameters. The proposed SBD can be also applied when there is no historical data but only prior information based on subject matter expertise.

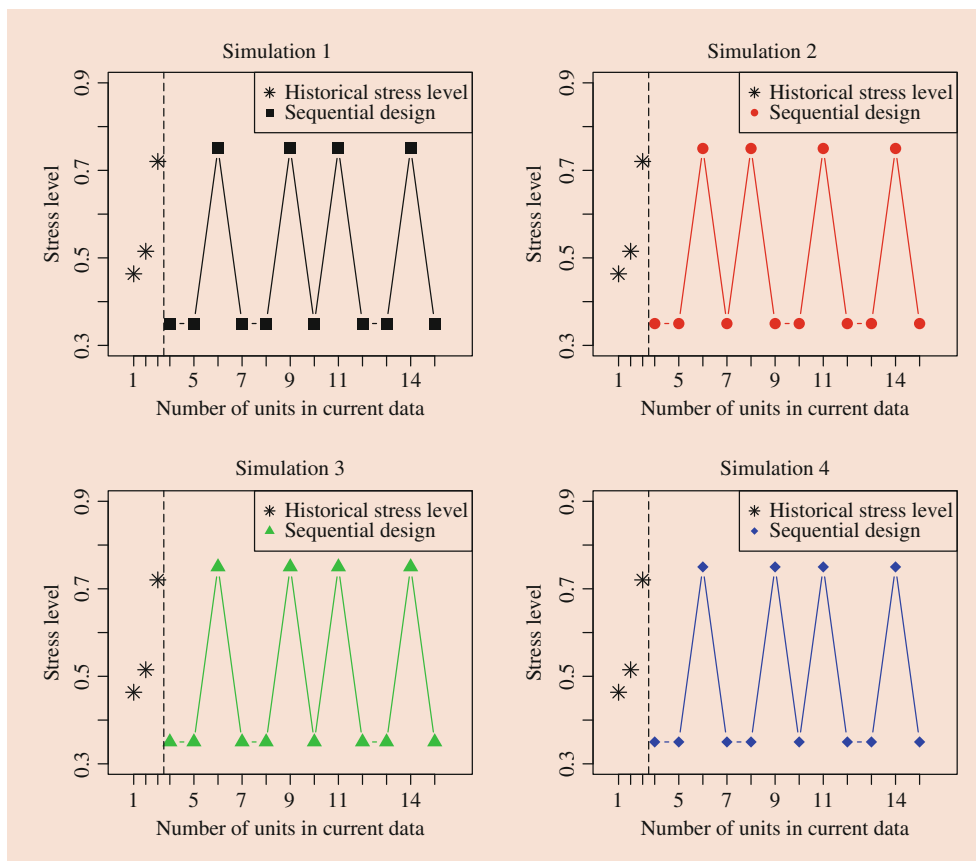


Fig. 6.19 Plots show the results of the four simulation trials including the sequential design points. *Figure reproduced with permission from Taylor and Francis*

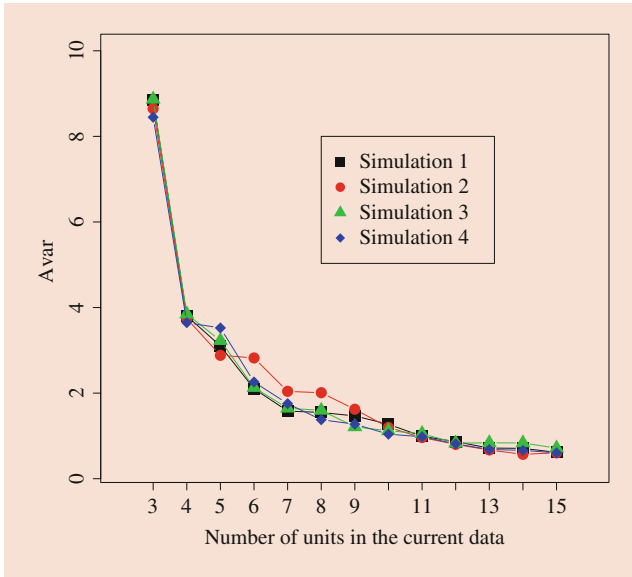


Fig. 6.20 The asymptotic variance of the four simulation trails. Figure reproduced with permission from Taylor and Francis

6.7 Sequential Test Planning Considering Dual Objectives

In the previous section, a sequential test plan was proposed based on the c -optimality, which focuses on obtaining precise prediction of the quantile lifetime at the normal use conditions. The D -optimality designs could be sought for when the focus is on obtaining precise estimates of the model parameters. The selection of optimal test plans requires specifying some planning values of the model parameters. It is often hard to precisely specify these parameter values at the early stage of the test when there is little information available about them. Hence, the chosen test plan based on the imprecisely specified parameter values might be suboptimal for the objective of interest. To quickly obtain good estimates of model parameters to guide more informative test planning, Lu et al. [44] propose a sequential test plan based on considering dual objectives. Particularly, it is recommended to consider D -optimality in the early stage of the sequential test plan to maximize the information gain on the model parameters and then followed by seeking c -optimality in the later stage to maximize the precision of the predictions at the normal use conditions. The idea is to use the early test units to maximally improve our knowledge on the unknown model parameters and then use later runs to gather more information on improving our capability of making predictions. The proposed dual objective sequential design is also applied to the polymer composites fatigue testing and is compared with alternative single objective test plans to demonstrate its performance [44].

6.7.1 Testing Criteria

When using D -optimality in the sequential framework, the criterion for selecting the $(n + 1)$ th run given the first n runs can be expressed as

$$\psi(q_{\text{new}}) = \int_{\Theta} [\log |I_{n+1}(\theta, q_{\text{new}})|] \pi(\theta | q_n, t_n, \delta_n) d\theta, \quad (6.29)$$

where $I_{n+1}(\theta, q_{\text{new}}) = I_n(\theta, q_n) + I_1(\theta, q_{\text{new}})$ and $|I_{n+1}(\theta, q_{\text{new}})|$ is the determinant of the Fisher information matrix based on the first $n + 1$ runs. Under the sequential framework, the total information gained from the first $n + 1$ runs can be updated from what was gained from the first n test units by adding the additional information obtained from testing the $(n + 1)$ th unit at q_{new} . Then, the optimal $(n + 1)$ th design point given the first n test units is determined by maximizing the total expected information gain under the posterior distribution of parameters conditioned on the first n runs as in

$$q_{n+1}^* = \arg \max_{q_{\text{new}} \in [q_L, q_U]} \psi(q_{\text{new}}). \quad (6.30)$$

Since the precision of model parameters is of major concern in the early stage of testing, the test units are determined based on optimizing the conditional information gain given the earlier runs via (6.30). When we have obtained sufficient information about the model parameters, the precision of predictions at the normal use condition is more of interest. Therefore, we switch to choosing further sequential runs by optimizing the asymptotic prediction variance in (6.28).

6.7.2 Procedure of Sequential Planning Based on Dual Objectives

We use N to denote the total number of test units used in the dual objective sequential test plan, and let N_1 and N_2 denote the units selected by the D -optimality and the c -optimality, respectively, where $N_1 + N_2 = N$. Similarly to the sequential test plan discussed in Sect. 6.6.3, the steps for sequential test planning based on considering dual objectives are summarized below:

1. *Specify prior distributions of model parameters.* For the polymer fatigue testing example, we specify prior distributions of A and B as $A \sim N(\mu_A, \sigma_A^2)$ and $B \sim N(\mu_B, \sigma_B^2)$, where μ_A , σ_A^2 , μ_B , and σ_B^2 are the hyperparameters of the normal prior distributions and are assumed to be known constants. Also we assume $v^2 \sim \text{Inverse Gamma}(\kappa, \gamma)$, where κ and γ are the known hyperparameters specified from the historical data or based on the subject matter expertise.

Table 6.1 Five sequential Bayesian designs. *Table reproduced with permission from Springer*

Design	Description
A: 12 <i>c</i> -opt	All 12 sequential runs are generated based on <i>c</i> -optimality
B: 12 <i>D</i> -opt	All 12 sequential runs are generated based on <i>D</i> -optimality
C: 6 <i>D</i> -opt + 6 <i>c</i> -opt	First 6 runs are obtained via <i>D</i> -optimality and last 6 runs via <i>c</i> -optimality
D: 4 <i>D</i> -opt + 8 <i>c</i> -opt	First 4 runs are obtained via <i>D</i> -optimality and last 8 runs via <i>c</i> -optimality
E: 2 <i>D</i> -opt + 10 <i>c</i> -opt	First 2 runs are obtained via <i>D</i> -optimality and last 10 runs via <i>c</i> -optimality

2. Evaluate the asymptotic variance. We recommend using the MCMC approach to approximate (6.29) and (6.27) for *D*-optimality and *c*-optimality, respectively.
3. Determine the optimal next testing point q_{n+1}^* at $(n + 1)$ th run. Given a candidate set of design points, their corresponding values of the objective function in (6.29) or (6.27) can be evaluated in step 2. Then, the optimal next design point is determined by optimizing the criterion over all the candidate points. If the criterion is *D*-optimality, then the optimum point is determined by (6.30). If the criterion is *c*-optimality, the optimum level is determined by (6.28).
4. Obtain the failure data at the selected level q_{n+1}^* . At the selected stress level q_{n+1}^* from step 3, conduct the experiment to obtain the failure information on (t_{n+1}, δ_{n+1}) .
5. Repeat steps 2 to 4 till the desired number of testing units is obtained. Add the new lifetime data at the $(n + 1)$ th run, $(q_{n+1}^*, t_{n+1}, \delta_{n+1})$, to the sequential test dataset. Repeat steps 2 to 4 till N_1 *D*-optimal test units and N_2 *c*-optimal runs are obtained, respectively.

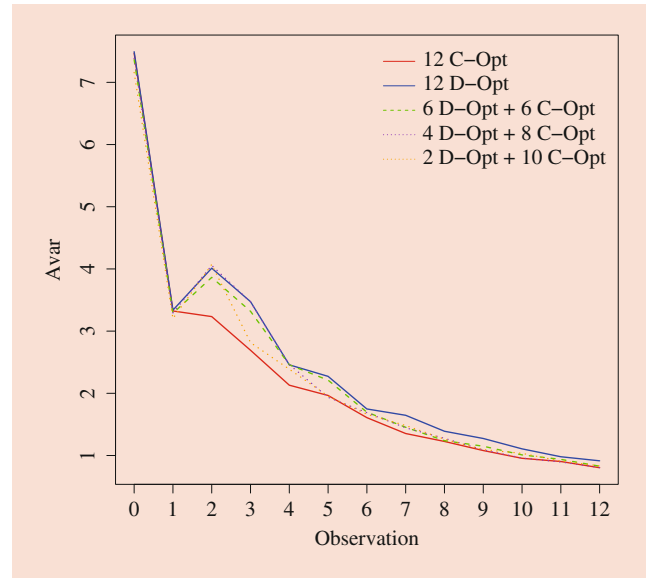


Fig. 6.21 Plot of AVar, the estimated asymptotic variance of the estimated quantile of lifetime averaged over a range of specified use condition as in (6.26) for the five designs from Table 6.1. *Figure reproduced with permission from Springer*

6.7.3 Polymer Composites Fatigue Testing

Here we revisit the example of polymer composites fatigue testing in Sect. 6.6.4. With the same settings of the prior information and historical data in Sect. 6.6.4, we also consider the scenario of collecting 12 new test units as in Sect. 6.6.4. To evaluate the proposed dual objective test plans, we compare five different test plans (summarized in Table 6.1) of the same sample size with two test plans obtained based on a single objective (either *D*- or *c*-optimality) and the remaining three being dual objective test plans with different sample size allocations among the *D*-optimal and *c*-optimal runs. Note that **Design A** is the *c*-optimal sequential test plan proposed in Sect. 6.6.

For each scenario, 100 simulation trials are generated and summarized to compare the performance on the asymptotic variance in (6.26) and the precision of estimated model parameters among the five test plans. To quantify the precision of estimation, the following metric of the relative precision of estimated model parameters is used:

$$m(\theta_j) = \frac{1}{K} \sum_{k=1}^K \left(\frac{|\hat{\theta}_{j,k} - \theta_j|}{\theta_j} \right)^2,$$

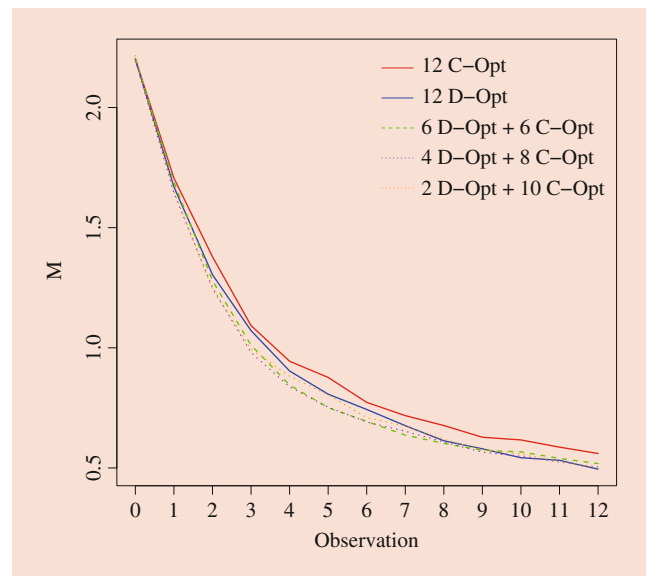


Fig. 6.22 Comparison of the *M* measure, the total relative mean square error for all three model parameters for the five designs from Table 6.1. *Figure reproduced with permission from Springer*

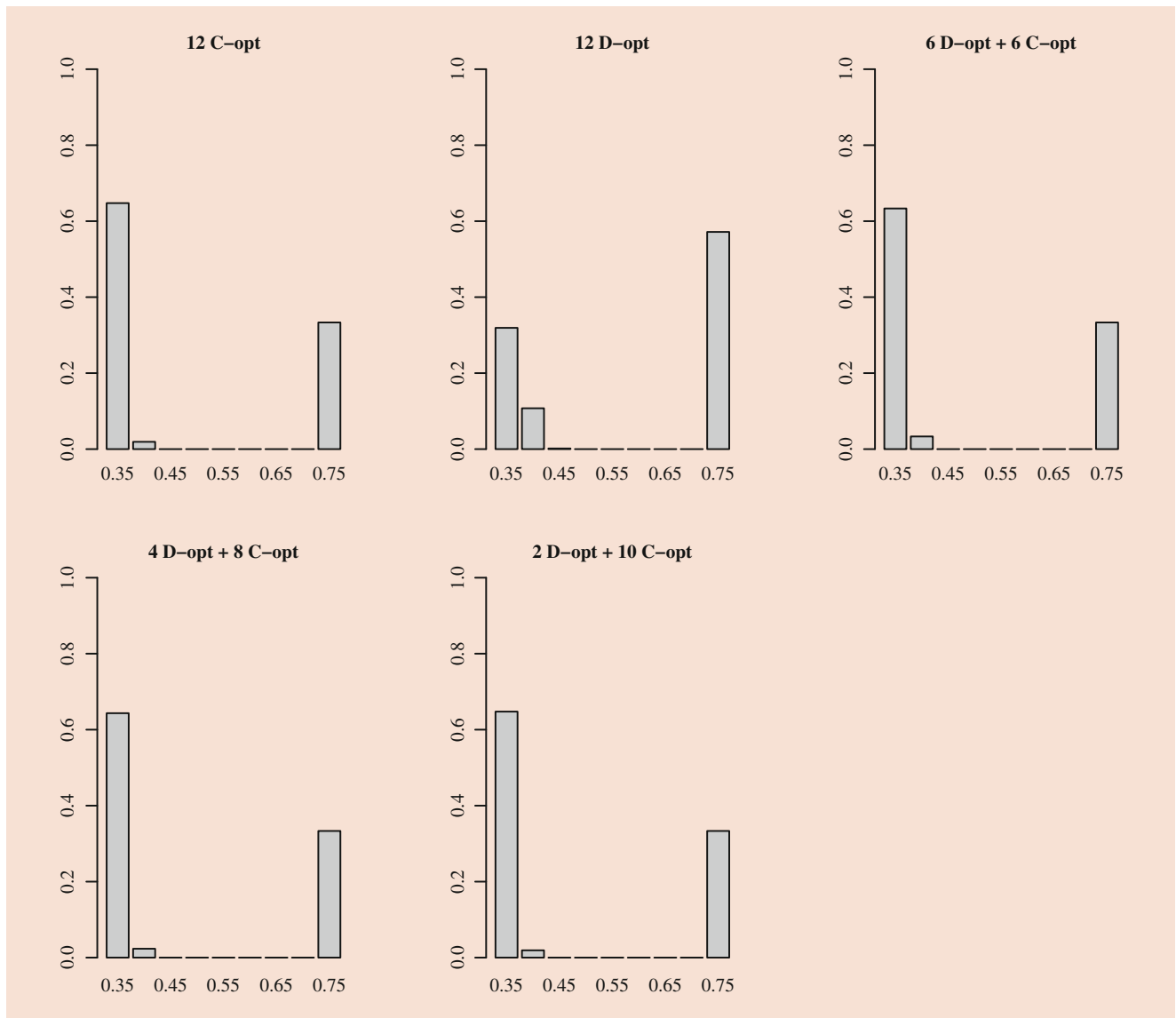


Fig. 6.23 Plot of sample size allocation for the five designs. *Figure reproduced with permission from Springer*

where θ_j represents the j th parameter of θ and $\hat{\theta}_{j,k}$ represents the estimates of θ_j from the k th trial for $k = 1, \dots, K = 100$. To summarize $m(\theta_j)$ across the three model parameters,

$$M = \sum_{j=1}^3 m(\theta_j)$$

was used to measure the total relative mean squared error of all the parameters. That is, M quantifies the overall precision of the estimates of all the model parameters in the ALT model. Note that the smaller value of M indicates overall more precisely estimated model parameters.

Figures 6.21 and 6.22 show the values of the asymptotic variance and M for the five scenarios, respectively. We can

see that **Design A** (12 units all generated by the c -optimality) has the smallest values of asymptotic variance $AVar$ and the largest values of M for all the runs. This indicates that **Design A** focuses more on obtaining most precisely predicted quantile of lifetime but less on the precision of estimated parameters. In addition, **Design B** (12 units all generated by the D -optimality) has the largest values of $AVar$ and relatively small M values for all the runs. Particularly, it achieves the best D -optimality after testing 8 units. This suggests that the D -optimal design does not provide the best predictions but instead offers more precise estimation of model parameters. However, the D -optimality is guaranteed at a relatively larger sample size.

Other three designs obtained by the dual objectives (**Designs C, D, and E**) generally perform well on the

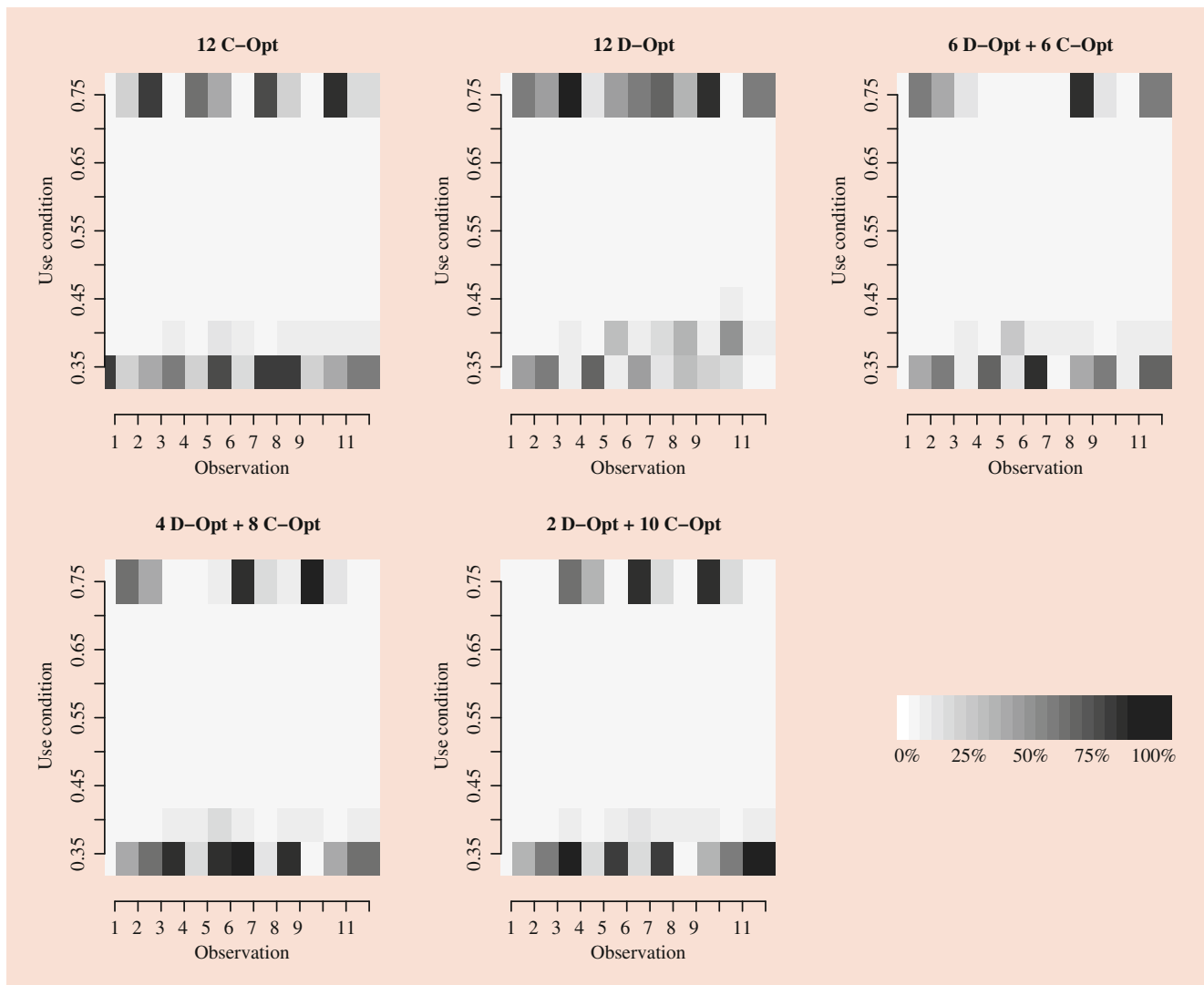


Fig. 6.24 Plot of the sample size allocation for the 12 sequential runs for the five designs. *Figure reproduced with permission from Springer*

both metrics, which offer more balanced performance between the asymptotic variance and the prediction of the estimated model parameters. Particularly, **Design D** with 4 D -optimality runs followed by 8 c -optimality runs has slightly better performance than other designs, which offers both the smallest $AVar$ and M . Based on the case study, the dual objective sequential designs outperform the D -optimality design if one is limited to 8 sequential runs. The dual objective sequential designs have better precision in terms of estimating model parameters and predicting quantile of lifetime.

Figure 6.23 shows the sample size allocation of the 12 sequential test units averaged over 100 simulation trials for the five test plans. In general, the test units are allocated to either the lowest or highest levels across the range of stress levels [0.35, 0.75]. The proportion of allocations to the lowest and highest stress levels for the D -optimality design is different from the c -optimality design and the dual objective

test plans. It is noted that there are twice as many units allocated to the highest level than the lowest level by the D -optimality test plan, while both c -optimality and the dual objective test plans allocate more test units to the lowest level. The c -optimality design has almost twice units tested at the lowest level than the highest level, while the dual objective designs provide similar allocation with a slightly larger sample size at the highest stress level. This is intuitive as the c -optimality emphasizes more on the prediction at the normal use condition; hence, testing more units at the lowest stress level with the closest proximity to the normal use condition would offer more useful information on improving the predictions.

Figure 6.24 shows the proportion of sample allocations for the five test plans. The plots show the allocations for each sequential run based on the 100 simulation trials. The light-to-dark gray shades indicate small to large allocation at the different testing levels. First, we can observe a large

middle region in the lightest gray shade for all the test plans, which indicates there is rarely a sample allocated to the middle levels (i.e., between $q = 0.4$ and $q = 0.7$) for all the scenarios. As compared to the c - and D -optimal plans, the D -optimality design allocates more runs at the highest stress level during the later runs, and for the earlier runs, the samples are evenly allocated between the highest and lowest levels. For the c -optimality design, it tends to allocate more units at the lower stress level during the later stage. This is because more samples at the higher stress levels can quickly improve the precision of estimated parameters. Testing units at the lower levels is more relevant for improving the prediction precision.

6.8 Concluding Remarks

In this chapter, we review recent developments on statistical reliability analysis utilizing dynamic covariates and sequential test planning. For time-to-event data, we introduce a cumulative damage model to account for the effect of dynamic covariates and illustrate the method with the Product D2 application. For degradation data, we present the general path model for incorporating dynamic covariates and illustrate the method with the NIST coating degradation data. We also introduce the MTRP model for recurrent events using dynamic covariates and illustrate it with the Vehicle B data. With regard to test planning for ALTs, we focus on the SBDs and illustrate it with the ALTs for polymer composites fatigue testing.

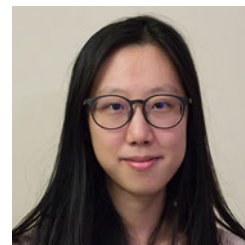
Looking forward, more versatile data become available due to the rapid advance of modern technology, and new statistical methods need to be developed to make use of those new data for improving reliability modeling and prediction. As described in [3], many data types such as spatial data, functional data, image data, and text data all have great potential to be used for reliability modeling and analysis. New methods that are available in statistical and machine learning can also be transformed and integrated with reliability domain knowledge for improving reliability analysis, which provides tremendous opportunity in reliability research.

Acknowledgments The authors thank the editor for his valuable comments which greatly helped improve this chapter. The research by Wang and Hong was partially supported by National Science Foundation Grants CMMI-1904165 to Virginia Tech.

References

1. Meeker, W.Q., Escobar, L.A.: *Statistical Methods for Reliability Data*. Wiley, New York (1998)
2. Meeker, W.Q., Hong, Y.: Reliability meets big data: opportunities and challenges, with discussion. *Qual. Eng.* **26**, 102–116 (2014)
3. Hong, Y., Zhang, M., Meeker, W.Q.: Big data and reliability applications: The complexity dimension. *J. Qual. Technol.* **50**(2), 135–149 (2018)
4. Lawless, J.F., Crowder, M.J., Lee, K.A.: Analysis of reliability and warranty claims in products with age and usage scales. *Technometrics* **51**, 14–24 (2009)
5. Guo, H., Monteforte, A., Mettas, A., Ogden, D.: Warranty prediction for products with random stresses and usages. In: *IEEE Proceedings Annual Reliability and Maintainability Symposium*, pp. 72–77. IEEE, Fort Worth, TX (2009)
6. Lu, L., Anderson-Cook, C.M.: Using age and usage for prediction of reliability of an arbitrary system from a finite population. *Qual. Reliab. Eng. Int.* **27**, 179–190 (2011)
7. Hong, Y., Meeker, W.Q.: Field-failure and warranty prediction based on auxiliary use-rate information. *Technometrics* **52**, 148–159 (2010)
8. Nelson, W.: Prediction of field reliability of units, each under differing dynamic stresses, from accelerated test data. In: Balakrishnan, N., Rao, C.R. (eds.) *Handbook of Statistics 20: Advances in Reliability*. North-Holland, Amsterdam (2001). Chap. IX
9. Voiculescu, S., Guérin, F., Barreau, M., Charki, A.: Reliability estimation in random environment: different approaches. In: *IEEE Proceedings Annual Reliability and Maintainability Symposium*, pp. 202–307. IEEE, Orlando, FL (2007)
10. Hong, Y., Meeker, W.Q.: Field-failure predictions based on failure-time data with dynamic covariate information. *Technometrics* **55**, 135–149 (2013)
11. Whitmore, G.A.: Estimation degradation by a Wiener diffusion process subject to measurement error. *Lifetime Data Anal.* **1**, 307–319 (1995)
12. Doksum, K.A., Hóyland, A.: Models for variable-stress accelerated life testing experiments based on Wiener processes and the inverse Gaussian distribution. *Technometrics* **34**, 74–82 (1992)
13. Wang, X.: Wiener processes with random effects for degradation data. *J. Multivar. Anal.* **101**, 340–351 (2010)
14. Lawless, J.F., Crowder, M.: Covariates and random effects in a gamma process model with application to degradation and failure. *Lifetime Data Anal.* **10**, 213–227 (2004)
15. Wang, X., Xu, D.: An inverse Gaussian process model for degradation data. *Technometrics* **52**, 188–197 (2010)
16. Ye, Z.-S., Chen, N.: The inverse Gaussian process as a degradation model. *Technometrics* **56**, 302–311 (2014)
17. Lu, C.J., Meeker, W.Q.: Using degradation measures to estimate a time-to-failure distribution. *Technometrics* **34**, 161–174 (1993)
18. Meeker, W.Q., Escobar, L.A., Lu, C.J.: Accelerated degradation tests: modeling and analysis. *Technometrics* **40**, 89–99 (1998)
19. Bagdonavičius, V., Nikulin, M.S.: Estimation in degradation models with explanatory variables. *Lifetime Data Anal.* **7**, 85–103 (2001)
20. Bae, S.J., Kuo, W., Kvam, P.H.: Degradation models and implied lifetime distributions. *Reliab. Eng. Syst. Saf.* **92**, 601–608 (2007)
21. Duan, Y., Hong, Y., Meeker, W., Stanley, D., Gu, X.: Photodegradation modeling based on laboratory accelerated test data and predictions under outdoor weathering for polymeric materials. *Ann. Appl. Stat.* **11**, 2052–2079 (2017)
22. Escobar, L.A., Meeker, W.Q., Kugler, D.L., Kramer, L.L.: Accelerated destructive degradation tests: data, models, and analysis. In: Lindqvist, B.H., Doksum, K.A. (eds.) *Mathematical and Statistical Methods in Reliability*. World Scientific Publishing Company, Singapore (2003)
23. Xie, Y., King, C.B., Hong, Y., Yang, Q.: Semi-parametric models for accelerated destructive degradation test data analysis. *Technometrics* **60**, 222–234 (2018)
24. Ding, Y., Yang, Q., King, C.B., Hong, Y.: A general accelerated destructive degradation testing model for reliability

- analysis. *IEEE Trans. Reliab.* **68**(4), 1272–1282 (2019). DOI: 10.1109/TR.2018.2883983
25. Hong, Y., Duan, Y., Meeker, W.Q., Stanley, D.L., Gu, X.: Statistical methods for degradation data with dynamic covariates information and an application to outdoor weathering data. *Technometrics* **57**, 180–193 (2015)
 26. Xu, Z., Hong, Y., Jin, R.: Nonlinear general path models for degradation data with dynamic covariates. *Appl. Stoch. Model. Bus. Ind.* **32**, 153–167 (2016)
 27. Zhao, R., Liu, B.: Renewal process with fuzzy interarrival times and rewards. *Int. J. Uncertainty Fuzziness Knowledge Based Syst.* **11**, 573–586 (2003)
 28. Hong, Y., Li, M., Osborn, B.: System unavailability analysis based on window-observed recurrent event data. *Appl. Stoch. Model. Bus. Ind.* **31**, 122–136 (2015)
 29. Kijima, M.: Some results for repairable systems with general repair. *J. Appl. Probab.* **26**, 89–102 (1989)
 30. Wang, H., Pham, H.: A quasi renewal process and its applications in imperfect maintenance. *Int. J. Syst. Sci.* **27**, 1055–1062 (1996)
 31. Doyen, L., Gaudoin, O.: Classes of imperfect repair models based on reduction of failure intensity or virtual age. *Reliab. Eng. Syst. Saf.* **84**, 45–56 (2004)
 32. Lindqvist, B., Elvebakk, G., Heggland, K.: The trend-renewal process for statistical analysis of repairable systems. *Technometrics* **45**, 31–44 (2003)
 33. Yang, Q., Hong, Y., Chen, Y., Shi, J.: Failure profile analysis of complex repairable systems with multiple failure modes. *IEEE Trans. Reliab.* **61**, 180–191 (2012)
 34. Pietzner, D., Wienke, A.: The trend-renewal process: a useful model for medical recurrence data. *Stat. Med.* **32**, 142–152 (2013)
 35. Yang, Q., Hong, Y., Zhang, N., Li, J.: A copula-based trend-renewal process model for analysis of repairable systems with multitype failures. *IEEE Trans. Reliab.* **66**(3), 590–602 (2017)
 36. Xu, Z., Hong, Y., Meeker, W.Q., Osborn, B.E., Illouz, K.: A multi-level trend-renewal process for modeling systems with recurrence data. *Technometrics* **59**, 225–236 (2017)
 37. Meeker, W.Q.: A comparison of accelerated life test plans for Weibull and lognormal distributions and type I censoring. *Technometrics* **26**(2), 157–171 (1984)
 38. Nelson, W.: *Accelerated Testing: Statistical Models, Test Plans, and Data Analyses*, (Republished in a paperback in Wiley Series in Probability and Statistics, 2004). Wiley, New York (1990)
 39. Zhang, Y., Meeker, W.Q.: Bayesian life test planning for the Weibull distribution with given shape parameter. *Metrika* **61**(3), 237–249 (2005)
 40. Zhang, Y., Meeker, W.Q.: Bayesian methods for planning accelerated life tests. *Technometrics* **48**(1), 49–60 (2006)
 41. Hong, Y., King, C.B., Zhang, Y., Meeker, W.Q.: Bayesian life test planning for log-location-scale family of distributions. *J. Qual. Technol.* **47**, 336–350 (2015)
 42. King, C., Hong, Y., Dehart, S.P., Defeo, P.A., Pan, R.: Planning Fatigue Tests for Polymer Composites. *J. Qual. Technol.* **48**, 227–245 (2016)
 43. Lee, I.-C., Hong, Y., Tseng, S.-T., Dasgupta, T.: Sequential Bayesian design for accelerated life tests. *Technometrics* **60**(4), 472–483 (2018)
 44. Lu, L., Lee, I., Hong, Y.: Bayesian Sequential design based on dual objectives for accelerated life tests. In: Lio, Y., Ng, H., Tsai, T., Chen, D. (eds.) *Statistical Quality Technologies*, pp. 257–277. Springer, Berlin (2019)
 45. Meeker, W.Q., Hong, Y., Escobar, L.A.: *Degradation Models and Analyses*. In: *Handbook of Engineering, Quality Control, and Physical Sciences*. Wiley, New York (2010). DOI: 10.1002/0471667196.ess7148
 46. Hong, Y., Meeker, W.Q., McCalley, J.D.: Prediction of remaining life of power transformers based on left truncated and right censored lifetime data. *Ann. Appl. Stat.* **3**, 857–879 (2009)
 47. Bagdonavičius, V., Nikulin, M.S.: *Accelerated Life Models: Modeling and Statistical Analysis*. Chapman & Hall/CRC, Boca Raton (2001)
 48. Lawless, J.F., Fredette, M.: Frequentist prediction intervals and predictive distributions. *Biometrika* **92**, 529–542 (2005)
 49. Hong, Y.: On Computing the distribution function for the Poisson Binomial Distribution. *Comput. Stat. Data Anal.* **59**, 41–51 (2013)
 50. Meyer, M.C.: Inference using shape-restricted regression splines. *Ann. Appl. Stat.* **2**, 1013–1033 (2008)
 51. Fraser, D., Massam, A.S.H.: A mixed primal-dual bases algorithm for regression under inequality constraints. Application to concave regression. *Scand. J. Stat.* **16**, 65–74 (1989)
 52. Carpenter, J.R., Goldstein, H., Rasbash, J.: A novel bootstrap procedure for assessing the relationship between class size and achievement. *Appl. Stat.* **52**, 431–443 (2003)
 53. Lütkepohl, H.: *New Introduction to Multiple Time Series Analysis*, 2nd edn. Springer, Berlin (2005)
 54. Epaarachchi, J.A., Clausen, P.D.: An empirical model for fatigue behavior prediction of glass fibre-reinforced plastic composites for various stress ratios and test frequencies. *Compos. A: Appl. Sci. Manuf.* **34**(4), 313–326 (2003)



Yueyao Wang received Ph.D. degree in Statistics in the Department of Statistics at Virginia Tech. Her research interests include computer experiments and reliability analysis.



I-Chen Lee received the Ph.D. in Statistics from National Tsing Hua University in 2016. She is an assistant professor in Department of Statistics at National Cheng Kung University. Her research interests include reliability analysis and optimal design.



Lu Lu received her PhD in Statistics from the Iowa State University. She is an Associate Professor of Statistics in the Department of Mathematics and Statistics at the University of South Florida. Her research areas include reliability data analysis, design of experiment, response surface methodology, multiple objective optimization and decision-making.



Yili Hong received the Ph.D. in Statistics from Iowa State University, in 2009. He is a Professor of Statistics at Virginia Tech. His research interests include machine learning and engineering applications, reliability analysis, and spatial statistics. He is currently an associate editor for *Technometrics* and *Journal of Quality Technology*. He is an elected member of International Statistical Institute.



Mathematical Reliability Aspects of Multivariate Probability Distributions

7

Lidia Z. Filus and Jerzy K. Filus

Contents

7.1	Introduction	129
7.2	General Structure of Any k -Variate Survival Function	130
7.3	The Main Result	132
7.4	Commentaries	134
	References	136

Abstract

This work deals with a comprehensive solution to the problem of finding the joint k -variate probability distributions of random vectors (X_1, \dots, X_k) , given all the univariate marginals. The general and universal analytic form of all solutions, given the fixed (but arbitrary) univariate marginals, was given in proven theorem. In order to choose among these solutions, one needs to determine proper “dependence functions” (joiners) that impose specific stochastic dependences among subsets of the set $\{X_1, \dots, X_k\}$ of the underlying random variables. Some methods of finding such dependence functions, given the fixed marginals, were discussed in our previous papers (Filus and Filus, *J Stat Sci Appl* 5:56–63, 2017; Filus and Filus, General method for construction of bivariate stochastic processes given two marginal processes. Presentation at 7-th International Conference on Risk Analysis, ICRA 7, Northeastern Illinois University, Chicago, 4 May 2017). In applications, such as system reliability modeling and

other, among all the available k -variate solutions, one needs to choose those that may fit particular data, and, after that, test the chosen models by proper statistical methods. The theoretical aspect of the main model, given by formula (7.3) in Sect. 7.2, mainly relies on the existence of one [for any fixed set of univariate marginals] general and universal form which plays the role of paradigm describing the whole class of the k -variate probability distributions for arbitrary $k = 2, 3, \dots$. An important fact is that the initial marginals are arbitrary and, in general, each may belong to a different class of probability distributions. Additional analysis and discussion are provided.

Keywords

Competitive to the copula methodology · Universal form of k -variate probability distributions · Stochastic dependences general structure · Joiners as “dependence functions” · r -independence · Reliability

7.1 Introduction

Consider a new approach to the old problem of finding a k -variate joint probability distributions of a random vector (X_1, \dots, X_k) , given its univariate marginals $S_1(x_1), \dots, S_k(x_k)$. (In this chapter, all the probability distributions will be represented by the corresponding survival functions. Nevertheless, we mostly will call them “distributions” for short, as they are equivalent to probability distributions. At this point also recall that in reliability theory the survival functions are called “reliability functions”). The problem, even if it often appears in literature (see, for example the references in [10]) with a variety of different formulations [3, 4, 8, 9, 11], is actually far from a comprehensive solution.

As an exception may serve the **copula** methodology by Sklar [12], which however has serious drawbacks that make

L. Z. Filus
Department of Mathematics, Northeastern Illinois University, Chicago, IL, USA
e-mail: L-Filus@neiu.edu

J. K. Filus (✉)
Department of Mathematics and Computer Science, Oakton Community College, Des Plaines, IL, USA

it difficult to use in most practical applications. The practitioners in their mutual discussions often indicate on the need for other methods of construction that would complement those by copulas. This need becomes especially vital when the dimension of the underlying random vector is higher than 2 or 3. Also the number of models created by use of copulas is rather limited.

In this work, we attempt to build a **new, competitive to the copula methodology**, theory of k -variate survival functions that has three important features. First, of theoretical nature, is providing the general structure (a *universal form*) of any k -variate survival function by means of the joiners [6, 7]. Second, the structure's form facilitates processes of constructions in various applications especially in **reliability** problems. As a result, a huge number of new bivariate and trivariate probability distributions can be found. The third feature of the here created theory is that the considered marginals $S_1(x_1), \dots, S_k(x_k)$ may belong to arbitrary classes of survival functions and that the classes, in general, may be different from each other. Thus, in the general case, some of the univariate marginals may be of the continuous type, some of the discrete, and some neither. The stochastic dependence structure that "connects" finite subsets $\{X_{i(1)}, \dots, X_{i(r)}\}$ ($r \leq k$) of the set of random variables $\{X_1, \dots, X_k\}$ provide the already mentioned dependence functions (joiners) of the r finite real variables $x_{i(1)}, \dots, x_{i(r)}$.

The created general theory starts with the bivariate case as developed in our previous papers [5–7].

In the beginning (in [5]), we started with bivariate models construction as an extension of the Aalen version [1] of the Cox model [3] of stochastic dependence towards construction of bivariate distributions.

Then we realized in [6, 7] that the so obtained construction method, in its full generality, may be formulated independently from the original Cox and the Aalen ideas.

As we have found, the construction can be based on the created notion of (bivariate) joiner.

It was then realized that in so doing one obtains a **universal representation** of any (!) bivariate survival function.

Namely, according to [5, 6], any bivariate survival function, say $S(x_1, x_2)$, can be represented by the factored (universal) form:

$$S(x_1, x_2) = S_1(x_1) S_2(x_2) J(x_1, x_2), \quad (7.0)$$

where $S_1(x_1)$ and $S_2(x_2)$ are the marginal survival functions and the function $J(x_1, x_2)$ determines the stochastic dependencies between the random variables X_1, X_2 is the "joiner."

Whenever a construction process was faced (with the initial data: the marginals $S_1(x_1), S_2(x_2)$ given), the problem reduces to finding a proper joiner $J(x_1, x_2)$ "connecting" the two marginals.

The methods of finding the function $J(x_1, x_2)$, by solving a proper integral equation or inequality, as well other criteria of "fitting" the joiner to the given marginals [2], were developed in our, mentioned above, previous papers. Some important bivariate models were immediately found. Others that require more effort were postponed to a future.

Anyway, the "bivariate part" of our theory can be considered sufficiently developed even if many important particular problems remain open.

The extension of this theory to k -dimensional ($k \geq 3$) cases was initiated in [7] with special emphasis on the $k = 3$ case. The general case was only sketched. Here, we concentrate on the general case for $k = 3, 4, \dots$.

The main result is formulated as theorem 1 and proven by mathematical induction with respect to k (see formula (7.3)).

In discussion that follows the proof, we show some methods that allow simplification of formula (7.3) by assuming that some of the underlying r -variate joiners ($r \leq k$) reduce to 1, which corresponds to "partial independence." Assuming all the r -variate (for $r \geq 3$) joiners are 1, we arrive at the notion of "bi-dependence" which means that all the dependencies reduce to those between pairs of random variables, and in so doing we obtain the corresponding formula (7.11) that determines simplified version of the general pattern (7.3). In the case of bi-dependence, which seems to be quite realistic, the task of construction reduces to finding proper bivariate joiners of (not necessarily all) pairs of underlying random variables (X_i, X_j) for $1 \leq i < j \leq k$.

Consequently, in bi-dependence case, the theory of general k -variate distributions essentially reduces to the, already developed, general theory of bivariate distributions.

All those properties, possibly, make the bi-variate and the general k -variate theories competitive to the copula methodology [2].

7.2 General Structure of Any k -Variate Survival Function

Now, we start with the analysis of the general cases for any k .

The essence of k -dimensional models' general structure ($k = 3, 4, \dots$) relies on recurrence transition from all its $(k - 1)$ -dimensional marginals to the constructed k -dimensional survival function.

Namely, the algebraic structure of any k -dimensional probability distribution $S(x_1, \dots, x_k)$ obeys the following factorization pattern:

$$\begin{aligned} S(x_1, \dots, x_k) &= J_{1, \dots, k}(x_1, \dots, x_{k-1}, x_k) S^{(1)}(x_2, x_3, \dots, x_k) \\ &S^{(2)}(x_1, x_3, \dots, x_k) \dots S^{(k-1)}(x_1, \dots, x_{k-2}, x_k) \\ &S^{(k)}(x_1, \dots, x_{k-1}) \end{aligned} \quad (7.1)$$

where for each $j = 1, \dots, k$ the $(k - 1)$ dimensional marginal survival function

$S^{(j)}(x_1, x_2, \dots, x_{j-1}, x_{j+1}, x_{j+2}, \dots, x_k)$ contains all the arguments x_1, \dots, x_k with exception of x_j .

The function $J_{12\dots k}(x_1, \dots, x_k)$ that we will call “**k-joiner**” determines stochastic “**k – dependence**” between all the $(k - 1)$ – dimensional random vectors

$$(X_1, X_2, \dots, X_{k-1}), (X_1, X_2, \dots, X_{k-2}, X_k), \dots, \\ (X_2, X_3, \dots, X_k).$$

For a, given in advance, survival function $S(x_1, \dots, x_k)$, the corresponding (unique) k -joiner simply is given by the formula that follows from (7.1):

$$J_{12\dots k}(x_1, \dots, x_k) = S(x_1, \dots, x_k) / \{S^{(1)}(x_2, x_3, \dots, x_k) \\ S^{(2)}(x_1, x_3, \dots, x_k) \dots S^{(k-1)}(x_1, \dots, x_{k-2}, x_k) \\ S^{(k)}(x_1, \dots, x_{k-1})\}. \quad (7.1^*)$$

For values of x_1, \dots, x_k that make the denominator in (7.1*) equal zero the k -joiner $J_{12\dots k}(x_1, \dots, x_k)$ is undefined, but in this case we set in (7.1) $S(x_1, \dots, x_k) = 0$ so the k -dimensional survival function in (7.1) is still well defined.

In the case $J_{12\dots k}(x_1, \dots, x_k) = 1$, for all points (x_1, \dots, x_k) , we say that the random vector (X_1, \dots, X_k) is **k-independent**.

The form (7.1) is universal, so every (!) survival function $S(x_1, \dots, x_k)$ can be represented like that. This rather strong statement is the conclusion from the following realization of the common arithmetic identity:

$$S(x_1, \dots, x_k) = [S(x_1, \dots, x_k) / \{S^{(1)}(x_2, x_3, \dots, x_k) \\ S^{(2)}(x_1, x_3, \dots, x_k) \dots S^{(k-1)}(x_1, \dots, x_{k-2}, x_k) \\ S^{(k)}(x_1, \dots, x_{k-1})\}] \{S^{(1)}(x_2, x_3, \dots, x_k) \\ S^{(2)}(x_1, x_3, \dots, x_k) \dots S^{(k-1)}(x_1, \dots, x_{k-2}, x_k) \\ S^{(k)}(x_1, \dots, x_{k-1})\} \quad (7.1^{**})$$

which in light of (7.1*) is another version of (7.1).

The identity (7.1**) is always true whenever the underlying fraction is well defined. Otherwise, we set (7.1**) equal zero which is consistent with determination of (7.1).

Suppose all the $(k - 1)$ – dimensional distributions $S^{(1)}(x_2, x_3, \dots, x_k)$, $S^{(2)}(x_1, x_3, \dots, x_k)$, $\dots, S^{(k-1)}(x_1, \dots, x_{k-2}, x_k)$, $S^{(k)}(x_1, \dots, x_{k-1})$ are given.

The question that arises [2] is:

For which functions $K(x_1, \dots, x_k)$ [that are only candidates for joiners], the product $K(x_1, \dots, x_k) S^{(1)}(x_2, x_3, \dots, x_k) S^{(2)}(x_1, x_3, \dots, x_k) \dots S^{(k-1)}(x_1, \dots, x_{k-2}, x_k) S^{(k)}(x_1, \dots, x_{k-1})$ is a valid k -dimensional survival function,

so that $K(x_1, \dots, x_k) = J_{12\dots k}(x_1, \dots, x_k)$ and formula (7.1) is satisfied for some survival function $S(x_1, \dots, x_k)$.

Solutions of this problem for $k = 2, 3, \dots$ yield solutions of the problem stated in the beginning of Sect. 7.1. To this, however, we need first to be able to solve the problem of finding joint distributions of all random vectors of $(k - 1)$ dimension. For this task, we need in turn to find or have all the joint distributions of dimension $(k - 2)$ and so on, until we encounter the univariate distributions ($k = 1$) that are given in advance.

The structure of the problem solving within here created new theory is then hierarchical. The essence of the structure is the recurrence formula (7.1) that describes the transition from $(k - 1)$ dimensions to k . However, when goes to the construction’s practice, we need to start at the lowest “level zero” where we choose the univariate distributions as given by $S_1(x_1), \dots, S_k(x_k)$ to work with. No new theory at level zero ($k = 1$) needs to be provided. Actual first step of this theory is the transition from $k = 1$ to $k = 2$. This means a need for a theory of forming bivariate distributions from any two univariates. This was the subject of our previous papers already cited.

Now, we start to derive the general formula for $S(x_1, \dots, x_k)$ that “exhibits” all the steps of the construction. The theory for $k = 2$ was (to a satisfactory degree) already developed in [5]. The case $k = 3$ was described in [7]. Since the recent position was submitted but not yet published, we repeat the main result here for 3-dimensional survival function’s universal representation:

$$S(x_1, x_2, x_3) = J_{123}(x_1, x_2, x_3) S^{(3)}(x_1, x_2) S^{(2)}(x_1, x_3) \\ S^{(1)}(x_2, x_3) \\ = J_{123}(x_1, x_2, x_3) [J_{12}(x_1, x_2) S_1(x_1) S_2(x_2)] \\ [J_{13}(x_1, x_3) S_1(x_1) S_3(x_3)] [J_{23}(x_2, x_3) S_2(x_2) \\ S_3(x_3)] \\ = J_{123}(x_1, x_2, x_3) J_{12}(x_1, x_2) J_{13}(x_1, x_3) \\ J_{23}(x_2, x_3) [S_1(x_1)]^2 [S_2(x_2)]^2 [S_3(x_3)]^2, \quad (7.2)$$

where for each pair (i, j) of subscripts, such that $1 \leq i < j \leq 3$, $S_{ij}(x_i, x_j)$ is bivariate distribution of the random vector (X_i, X_j) so it is a bivariate marginal of the tri-variate distribution.

$S(x_1, x_2, x_3)$ of the random vector (X_1, X_2, X_3) , and, for each bivariate distribution $S_{ij}(x_i, x_j)$, $J_{ij}(x_i, x_j)$ denotes the corresponding joiner which “ties” the univariate marginal distributions $S_i(x_i), S_j(x_j)$. In (7.2) we denoted by $S^{(i)}(x_k, x_l)$ the bivariate survival function of those two arguments among x_1, x_2, x_3 that are different from x_i , for $i = 1, 2, 3$.

7.3 The Main Result

The analogical to (7.2) formula for an arbitrary dimension $k \geq 3$ is given by the following theorem, which will be proven by mathematical induction with respect to k .

Theorem 1 Suppose a sequence of random vectors $\{(X_1, X_2, \dots, X_{k-1}, X_k)\}_{k=1}^\infty$ is consistent in the sense of Daniell-Kolmogorov Consistency Theorem (however, the problem of symmetry of the random vectors is not analyzed here).

Under that assumption, for every $k = 2, 3, \dots$ the **“universal” form** of any survival function $S(x_1, x_2, \dots, x_{k-1}, x_k)$ of an arbitrary random vector $(X_1, X_2, \dots, X_{k-1}, X_k)$, whose marginals $X_1, X_2, \dots, X_{k-1}, X_k$ take on finite real values, is given by the following formula:

$$\begin{aligned} S(x_1, x_2, \dots, x_{k-1}, x_k) &= J_{1, \dots, k}(x_1, \dots, x_{k-1}, x_k) S^{(1)}(x_2, x_3, \dots, x_k) \\ &S^{(2)}(x_1, x_3, \dots, x_k) \dots S^{(k-1)}(x_1, \dots, x_{k-2}, x_k) S^{(k)}(x_1, \dots, x_{k-1}) \\ &= J_{1, \dots, k}(x_1, \dots, x_{k-1}, x_k) \prod_{s=1}^{k-1} \prod_{(j(1), \dots, j(k-s))}^{k-1} \\ &\left[J_{(j(1), \dots, j(k-s))}(x_{j(1)}, \dots, x_{j(k-s)}) \right]^{s!} \quad \square \end{aligned} \quad (7.3)$$

As for the middle expression in equalities (7.3), $J_{1, \dots, k}(x_1, \dots, x_{k-1}, x_k)$ is the k -joiner introduced above and (for $j = 1, 2, \dots, k$) $S^{(j)}(\dots)$ are $(k-1)$ -dimensional marginal survival functions of the random vector $(X_1, X_2, \dots, X_{k-1}, X_k)$ obtained from its survival function $S(x_1, x_2, \dots, x_{k-1}, x_k)$ by setting $x_j = 0$.

In the last expression of (7.3) symbol, $\prod_{s=1}^{k-1}$ means the product taken from $s = 1$ to $s = k-1$, while, for each particular s , the symbol $\prod_{(j(1), \dots, j(k-s))}^{k-1}$ denotes product of all expressions (joiners) each being a function of some $(k-s)$ arguments $x_{j(1)}, x_{j(2)}, \dots, x_{j(k-s)}$ only, where:

$$1 \leq j(1) < j(2) < \dots < j(k-s) \leq k, \text{ for some } s.$$

The symbol $J_{(j(1), \dots, j(k-s))}(x_{j(1)}, \dots, x_{j(k-s)})$ denotes the $(k-s)$ -dimensional joiner for the (marginal) random vector $(X_{j(1)}, \dots, X_{j(k-s)})$. Because of the consistency assumption, this joiner is raised to the power $s!$.

To achieve more uniform notation, for $s = k-1$ we introduced the notation

$$\begin{aligned} J_{(j(1), \dots, j(k-s))}(\dots) &= J_{j(1)}(x_{j(1)}) = S_{j(1)}(x_{j(1)}) = S_r(x_r) \\ &\text{for some } r = j(1) = 1, \dots, k. \end{aligned}$$

Thus, the so obtained “one-dimensional joiner” actually is not a joiner but rather the one-dimensional survival function. Realize that these survival functions are raised to the power $(k-1)!$ so that the product

$S_1(x_1)^{(k-1)!} \dots S_k(x_k)^{(k-1)!}$ is always the factor of $S(x_1, x_2, \dots, x_{k-1}, x_k)$ given by (7.3).

Proof of the Theorem We set $k = m+1$ and prove the theorem by mathematical induction with respect to m . According to formulas (7.0) and (7.2), the theorem holds for $m \leq 2$.

Now, suppose that it holds for some arbitrary m .

This means that similarly as in (7.3) we have:

$$\begin{aligned} S(x_1, x_2, \dots, x_{m-1}, x_m) &= J_{1, \dots, m}(x_1, \dots, x_{m-1}, x_m) S^{(1)}(x_2, x_3, \dots, x_m) \\ &S^{(2)}(x_1, x_3, \dots, x_m) \dots S^{(m-1)}(x_1, \dots, x_{m-2}, x_m) \\ &S^{(m)}(x_1, \dots, x_{m-1}) \\ &= J_{1, \dots, m}(x_1, \dots, x_{m-1}, x_m) \prod_{s=1}^{m-1} \\ &\prod_{(j(1), \dots, j(m-s))} J_{(j(1), \dots, j(m-s))}^{s!}(x_{j(1)}, \dots, x_{j(m-s)}), \end{aligned} \quad (7.4)$$

where $1 \leq j(1) < j(2) < \dots < j(m-s) \leq m; s = 1, \dots, m-1$.

According to defining formulas (7.1) and (7.1*), we also have

$$\begin{aligned} S^+(x_1, x_2, \dots, x_m, x_{m+1}) &= (J_{1, \dots, m+1}(x_1, \dots, x_m, x_{m+1})^+ \\ &S^{(1)}(x_2, x_3, \dots, x_{m+1})^+ S^{(2)}(x_1, x_3, \dots, x_{m+1})^+ \\ &\dots + S^{(m)}(x_1, \dots, x_{m-1}, x_{m+1})^+ S^{(m+1)}(x_1, \dots, x_m))^+ \end{aligned} \quad (7.5)$$

where $J_{1, \dots, m+1}(x_1, \dots, x_m, x_{m+1})$ is the $(m+1)$ -joiner for the whole random vector $(X_1, \dots, X_m, X_{m+1})$.

As for the remaining factors on the right hand side of (7.5), the marginal factor $+S^{(m+1)}(x_1, \dots, x_m)$ is (by the Kolmogorov consistency assumption that we adopted) identical to the function $S(x_1, x_2, \dots, x_{m-1}, x_m)$ given by (7.4).

[As the conclusion of that consistency restriction, any $(m+1)$ -dimensional survival function has the product form:

$$\begin{aligned} S^+(x_1, x_2, \dots, x_m, x_{m+1}) &= S(x_1, x_2, \dots, x_m) \\ &\Phi(x_1, x_2, \dots, x_m, x_{m+1}) \end{aligned} \quad]$$

The remaining factors $+S^{(i)}(i = 1, \dots, m)$ in (7.5) must obey the scheme given by (7.3) for any set of m real arguments. Therefore, they are products of the powers $J_{(j(1), \dots, j(m-s))}^{s!}(x_{j(1)}, \dots, x_{j(m-s)})$ of the underlying joiners having nearly the same syntax structure as that of $S(x_1, x_2, \dots, x_{m-1}, x_m)$ given by (7.4).

All the joiners present in the marginals $+S^{(i)}()$ that do not contain the argument $x_{j(m-s)} = x_{m+1}$ are, by the consistency assumption, exactly the same as the corresponding factors of $S(x_1, x_2, \dots, x_{m-1}, x_m)$ whenever they have the same arguments.

Other factors that contain the (“new”) argument x_{m+1} [such as, for example, $J_{2,m+1}(x_2, x_{m+1})$] differ from those which form the expression for $S(x_1, x_2, \dots, x_{m-1}, x_m)$.

Consider again $(m + 1)$ of m -variate marginals of $S^+(x_1, x_2, \dots, x_m, x_{m+1})$, present in (7.5).

Realize that every single variable x_i ($i = 1, 2, \dots, m + 1$) occurs in exactly m out of $m + 1$ marginals:

$$\begin{aligned} & +S^{(1)}(x_2, x_3, \dots, x_{m+1}), +S^{(2)}(x_1, x_3, \dots, x_{m+1}), \\ & \dots, +S^{(m)}(x_1, \dots, x_{m-1}, x_{m+1}), +S^{(m+1)}(x_1, \dots, x_m) \quad (\text{M}) \end{aligned}$$

The above functions are distinct, and each of them contains all the variables $x_1, x_2, \dots, x_m, x_{m+1}$ with exception of exactly one.

Consequently, each variable x_i ($i = 1, \dots, m + 1$) occurs in exactly m out of $m + 1$ functions in (M).

The functions given by (M) represent m -dimensional survival functions. Thus, according to (7.4), for a given i , each such function contains the factor $[S_i(x_i)]^{(m-1)!}$. This factor repeats m -times in the product

$$\begin{aligned} & +S^{(1)}(x_2, x_3, \dots, x_{m+1}) + S^{(2)}(x_1, x_3, \dots, x_{m+1}) \\ & \dots + S^{(m)}(x_1, \dots, x_{m-1}, x_{m+1}) + S^{(m+1)}(x_1, \dots, x_m) \end{aligned}$$

and therefore is raised to the power m , which results in the factor

$$[S_i(x_i)]^m \text{ of } S^+(x_1, x_2, \dots, x_m, x_{m+1}).$$

Since above reasoning is valid for every i , the product $[S_1(x_1)]^m \dots [S_{m+1}(x_{m+1})]^m$ is a factor of $S^+(x_1, x_2, \dots, x_m, x_{m+1})$.

Recall, the corresponding factor of $S(x_1, x_2, \dots, x_{m-1}, x_m)$ was the product $[S_1(x_1)]^{(m-1)!} \dots [S_m(x_m)]^{(m-1)!}$ as in the first product in (7.4) we have $1 \leq s \leq m - 1$.

Before we consider 2-dimensional factors (i.e., all the 2-dimensional joiners in (7.5*) below), we provide in advance the (final) formula for the $(m + 1)$ -dimensional case, which is the continuation of (7.5):

$$\begin{aligned} & S^+(x_1, x_2, \dots, x_m, x_{m+1}) \\ & = J_{1, \dots, m+1}(x_1, \dots, x_m, x_{m+1}) S^{(1)}(x_2, x_3, \dots, x_{m+1}) \\ & \quad S^{(2)}(x_1, x_3, \dots, x_{m+1}) \dots S^{(m)}(x_1, \dots, x_{m-1}, x_{m+1}) \\ & \quad S^{(m+1)}(x_1, \dots, x_m) \\ & = J_{1, \dots, m+1}(x_1, \dots, x_m, x_{m+1}) \\ & \quad \prod_{s=1}^m \prod_{(j(1), \dots, j(m+1-s))} J_{(j(1), \dots, j(m+1-s))} \\ & \quad s!(x_{j(1)}, \dots, x_{j(m+1-s)}), \end{aligned} \quad (7.5^*)$$

where $1 \leq j(1) < j(2) < \dots < j(m + 1 - s) \leq m + 1$.

Now, consider all the bivariate joiners $J_{(j(1), j(2))}(x_{j(1)}, x_{j(2)})$ present in the last expression of (7.4). Similarly, as we proceeded with 1-dimensional joiners (the marginal distributions) realize that [for $s = m - 2$] in (7.4), the bivariate joiners $J_{(j(1), j(2))}(x_{j(1)}, x_{j(2)})$ are all raised to the power $(m - 2)!$.

Roughly the same happens for all the factors

$$\begin{aligned} & S^{(1)}(x_2, x_3, \dots, x_{m+1}) S^{(2)}(x_1, x_3, \dots, x_{m+1}) \dots \\ & S^{(m)}(x_1, \dots, x_{m-1}, x_{m+1}) S^{(m+1)}(x_1, \dots, x_m) \text{ of } (7.5^*). \end{aligned}$$

Thus, in (7.5*) each two-dimensional expression, $J_{(j(1), j(2))}(x_{j(1)}, x_{j(2)})^{(m-2)!}$ is present $(m - 1)$ times [($m + 1$) times but two] as the factors. This is exactly the same as raising it to the power $m - 1$. Thus, as formula (7.4) transforms to formula (7.5*) [$m \rightarrow m + 1$], the factor $J_{(j(1), j(2))}(x_{j(1)}, x_{j(2)})^{(m-2)!}$ transforms to the factor

$$J_{(j(1), j(2))}(x_{j(1)}, x_{j(2)})^{(m-1)!}$$

(in the last case, however, we have $1 \leq j(1) < j(2) \leq m + 1$).

In the same way, one can show that since the factors $J_{(j(1), j(2), j(3))}(x_{j(1)}, x_{j(2)}, x_{j(3)})^{(m-3)!}$ in (7.4) are present in (7.5*) $(m - 2)$ times (again: $m + 1$ times but three) the factor $J_{(j(1), j(2), j(3))}(x_{j(1)}, x_{j(2)}, x_{j(3)})^{(m-3)!}$ transforms (as $m \rightarrow m + 1$) into the factor $J_{(j(1), j(2), j(3))}(x_{j(1)}, x_{j(2)}, x_{j(3)})^{(m-2)!}$.

Doing this successively in the same way, we see that, as the dimension raises from m to $m + 1$, every factor $J_{(j(1), \dots, j(r))}(x_{j(1)}, x_{j(2)}, \dots, x_{j(r)})^{(m-r)!}$ in (7.4) transforms into the factor $J_{(j(1), \dots, j(r))}(x_{j(1)}, x_{j(2)}, \dots, x_{j(r)})^{(m-r+1)!}$, in (7.5*), where $r = 1, 2, \dots, m$.

The single factor $J_{1, \dots, m+1}(x_1, \dots, x_m, x_{m+1})$ in (7.5*) plays the same role as the single factor $J_{1, \dots, m}(x_1, \dots, x_{m-1}, x_m)$ in (7.4).

Concluding, the survival function $S^+(x_1, x_2, \dots, x_m, x_{m+1})$ given by (7.5*) one can obtain from the survival function $S(x_1, x_2, \dots, x_{m-1}, x_m)$ given by (7.4) just by replacing all the indexes $1 \leq j(1) < j(2) < \dots < j(m - s) \leq m$ with the indexes $1 \leq j'(1) < j'(2) < \dots < j'(m + 1 - s) \leq m + 1$ and all the exponents [with the same basis] $(n - r)!$ to the exponents $(n - r + 1)!$. Without going too far into details one may conclude that the syntax form of the survival function $S^+(x_1, x_2, \dots, x_m, x_{m+1})$ is “the same” as that of $S(x_1, x_2, \dots, x_{m-1}, x_m)$, and therefore, the formulas (7.4) and (7.5*) are basically “the same.” Thus, the pattern for the survival function given by (7.4) naturally transforms into the same pattern given by (7.5*).

By mathematical induction, we conclude [upon $k - 1 = m$] that formula (7.3) holds for any $k = 2, 3, \dots$ \square

7.4 Commentaries

Remark 1 For more clarification, in addition to the above proof, notice that the m -dimensional factor $J_{1,\dots,m}(x_1, \dots, x_{m-1}, x_m)$, present only once in (7.4), will be “replaced” in (7.5*) by the product of $m + 1$ distinct factors $J_{(j(1), \dots, j(m))}(x_{j(1)}, x_{j(2)}, \dots, x_{j(m)})$ each of them “raised” to the power 1.

This time, however, $j^*(m) \leq m + 1$.

Hypothetically, if [as a possible “next step”] the random vector dimension would be increased from $m + 1$ to $m + 2$, all the factors corresponding to those m -dimensional would be raised to second (2!) power and next (for $m + 3$ – dimension) to 3! power, and so on.

Thus, for a given random vector $(X_1, \dots, X_m, X_{m+1})$, formula (7.5*) determines its joint survival function, given the joiners of all the [marginal] random vectors $(X_{j(1)}, X_{j(2)}, \dots, X_{j(r)})$, $r = 1, 2, \dots, m$ and one $(m + 1)$ - dimensional joiner $J_{1,\dots,m+1}(x_1, \dots, x_m, x_{m+1})$ which stands for a single factor in (7.5*). \square

Remark 2 For formula (7.5*) to hold, the assumption of the Kolmogorov consistency is the substantial one. Otherwise, for $s = 1, \dots, m - 1$, $m = 2, 3, \dots$ the joiners $J_{(j(1), \dots, j(m-s))}(x_{j(1)}, \dots, x_{j(m-s)})$ with the same arguments could be changed after each transition from m to $m + 1$. Therefore, instead of the powers $J_{(j(1), \dots, j(m+1-s))}^{s!}$, one could encounter fairly long products of distinct factors. The construction of k -variate survival function in such a case would still be possible but the underlying formulas would be much longer. In the 2-dimensional case (see, [7]), this “inconsistency phenomena” is possibly related to two different representations [marginal and baseline] of the same bivariate distribution.

In that, two single random variables are differently distributed when considered separately (baseline distributions) and in bivariate distribution (as the marginals). \square

Because of the complexity of formula (7.3) which is somewhat unintuitive, in our opinion formula (7.2) is not yet sufficient illustration of the pattern of the general k -dimensional survival function. To better exhibit the mechanism of “forming” k -dimensional distribution from a $(k - 1)$ -dimensional, we decided to provide additional example of forming 4-dimensional survival function from the 3-dimensionals whose forms are given by (7.2).

Example In accordance with (7.5) (for $m = 3$) the general form, a 4-dimensional survival function can be expressed by the product of the common 4-joiner and four 3-dimensional marginals. Thus, we have:

$$\begin{aligned} S(x_1, x_2, x_3, x_4) &= \\ &= J_{1234}(x_1, x_2, x_3, x_4) \{S_{123}(x_1, x_2, x_3)\} \{S_{124}(x_1, x_2, x_4)\} \\ &\quad \{S_{134}(x_1, x_3, x_4)\} \{S_{234}(x_2, x_3, x_4)\}. \end{aligned} \quad (7.6)$$

In accordance with the Kolmogorov consistency condition, the factor $S_{123}(x_1, x_2, x_3)$ in (7.6) is the same as $S(x_1, x_2, x_3)$ given by (7.2). Other marginal factors containing arguments different from the triple (x_1, x_2, x_3) are, in general, different. However, their syntax structure (i.e., as long as only the names of the underlying functions in the expressions are considered) is always the same.

Expanding each of the expressions of (7.6) in the parenthesis according to pattern (7.2), one obtains:

$$\begin{aligned} S(x_1, x_2, x_3, x_4) &= J_{1234}(x_1, x_2, x_3, x_4) \left\{ J_{123}(x_1, x_2, x_3) J_{12}(x_1, x_2) \right. \\ &\quad \left. J_{13}((x_1, x_3) J_{23}(x_2, x_3) [S_1(x_1)]^2 [S_2(x_2)]^2 [S_3(x_3)]^2 \right\} \\ &\quad \left\{ J_{124}(x_1, x_2, x_4) J_{12}(x_1, x_2) J_{14}(x_1, x_4) J_{24}(x_2, x_4) \right. \\ &\quad \left. [S_1(x_1)]^2 [S_2(x_2)]^2 [S_4(x_4)]^2 \right\} \\ &\quad \left\{ J_{134}(x_1, x_3, x_4) J_{13}(x_1, x_3) J_{14}(x_1, x_4) J_{34}(x_3, x_4) \right. \\ &\quad \left. [S_1(x_1)]^2 [S_3(x_3)]^2 [S_4(x_4)]^2 \right\} \\ &\quad \left\{ J_{234}(x_2, x_3, x_4) J_{23}(x_2, x_3) J_{24}(x_2, x_4) J_{34}(x_3, x_4) \right. \\ &\quad \left. [S_2(x_2)]^2 [S_3(x_3)]^2 [S_4(x_4)]^2 \right\} \end{aligned} \quad (7.7)$$

After regrouping all the factors in (7.7) and matching repeating factors that occur in different parentheses, we obtain the following formula:

$$\begin{aligned} S(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4) &= J_{1234}(x_1, x_2, x_3, x_4) J_{123}(x_1, x_2, x_3) \\ &\quad J_{124}(x_1, x_2, x_4) J_{134}(x_1, x_3, x_4) J_{234}(x_2, x_3, x_4) \\ &\quad [J_{12}(x_1, x_2)]^2 [J_{13}(x_1, x_3)]^2 [J_{14}(x_1, x_4)]^2 [J_{23}(x_2, x_3)]^2 \\ &\quad [J_{24}(x_2, x_4)]^2 [J_{34}(x_3, x_4)]^2 [S_1(x_1)]^6 [S_2(x_2)]^6 \\ &\quad [S_3(x_3)]^6 [S_4(x_4)]^6 \end{aligned} \quad (7.8)$$

Formula (7.8) seems not to be so complicated, but it is pretty long and the underlying (6-th) powers of the univariate marginals could make us somewhat uncomfortable. And this is only the 4-dimensional case. Raising this dimension only by one [see formula (7.5*) and proof of theorem 1] causes raising the 1-dimensional marginals to the power 24 and all the bivariate joiners to the power 6. That may suggest that such formulas are useless from a practical (reliability, for example) point of view, and what remains would be only the theoretical value of formula (7.5*). This “first impression,” however, is not well grounded. To see this only recall the defining formulae (7.1), (7.1*) and (7.1**). Thus, however small the factors $J_{(j(1), \dots, j(m+1-s))}^{s!}(x_{j(1)}, \dots, x_{j(m+1-s)})$ are, the products forming the corresponding value of the joiner’s $J_{1,\dots,k}(x_1, \dots, x_{k-1}, x_k)$ denominator [see (7.1*)] is proportionally high (much greater than 1).

As the net result, one always obtains a reasonable value, that is, the joint survival function $S(x_1, x_2, \dots, x_{k-1}, x_k)$, we analyzed (see formula (7.1**) and its further expansion (7.3)). \square

In vast majority of “regular” cases, formulas (7.8) and (7.3), in their full extend, are unrealistic especially whenever applications are of interest. Fortunately, the results as given by those formulas are just the first step that comprises the full generality which is seldom met in practice. Obviously, in applications, it is not too realistic situation when “everything depends on everything else.”

In reality (including also “most of” the “mathematical reality”), the vast majority of the joiners (present in (7.5*) and in (7.8)) should be considered to be equal to 1, which means corresponding stochastic independence.

The important theoretical (and practical as well) fact is that all the stochastic dependences (as well as independence) among the random variables X_1, \dots, X_k are entirely described by the set of all the joiners

$$J_{(j(1), \dots, j(r))} (x_{j(1)}, \dots, x_{j(r)}), \text{ where } r = 2, 3, \dots, k.$$

Thus, the dependences’ analysis mathematically reduces to analysis of the functions $J_{(j(1), \dots, j(r))} (x_{j(1)}, \dots, x_{j(r)})$ which fully represent the corresponding stochastic dependence. One can say they play a role similar to that of having only numerical values “correlation coefficients” defined in literature in variety of ways.

Remark 3 At this point is good to notice that the k -joiner $J_{1, \dots, k} (x_1, \dots, x_{k-1}, x_k)$ describes exclusively stochastic dependence of the set of all the $(k - 1)$ -dimensional random vectors

$$\left\{ (X_{j(1)}, \dots, X_{j(k-1)}) \right\}, \text{ where } 1 \leq j(1) < j(2) < \dots < j(k-1) \leq k.$$

This joiner “is responsible” only for the dependence between these $(k - 1)$ -dimensional random vectors. If it equals 1 (which means the **k-independence**; see in above) then, in accordance with (7.3), we have

$$\begin{aligned} S(x_1, x_2, \dots, x_{k-1}, x_k) \\ = S^{(1)}(x_2, x_3, \dots, x_k) S^{(2)}(x_1, x_3, \dots, x_k) \dots \\ S^{(k-1)}(x_1, \dots, x_{k-2}, x_k) S^{(k)}(x_1, \dots, x_{k-1}) \end{aligned} \quad (7.9)$$

so that the survival functions of all the $(k - 1)$ -dimensional marginals simply multiply by each other.

This factorization agrees with the common intuition of stochastic independence’ analytic descriptions.

It may happen that some disjoint proper subsets of the set $\{X_{j(1)}, \dots, X_{j(k-1)}\}$ that form random (sub)vectors marginal to the corresponding $(k - 1)$ -dimensional marginal random

vector $(X_{j(1)}, \dots, X_{j(k-1)})$ are *independent* from each other, while their corresponding (“inner”) “ k -sub-joiners” [factors of the k -joiner] are not equal to 1.

This phenomenon may be called “group independence,” where random variables within each such “group” [subset] may be dependent.

Returning to the general case of the joiners (factored or not), notice that any r -joiner $J_{(j(1), \dots, j(r))} (x_{j(1)}, \dots, x_{j(r)})$ of any r -dimensional marginal random vector $(X_{j(1)}, \dots, X_{j(r)})$ ($r \geq 2$) only describes dependences between the $(r - 1)$ -dimensional marginals of this vector.

The considered r -joiner may be factored or not according to existence or nonexistence of independence between some disjoint proper subsets of the associated sets $\{X_{j(1)}, \dots, X_{j(r)}\}$ that form at most $(r-1)$ -dimensional marginal random sub-vectors of the random vectors $(X_{j(1)}, \dots, X_{j(r)})$.

In particular, if the r -joiner is factored, some of the factors may be equal 1. \square

As already mentioned, in practical reality the number of dependent random subvectors (i.e., r -dimensional [$r \leq k - 1$] marginals) of the random vector (X_1, \dots, X_k) is rather limited.

This means most of the underlying r -variate joiners present in formulas (7.3) or (7.8) are simply equal to 1.

Regarding that phenomenon, the defined below notion of r -independence may be useful in various specific model constructions.

We finalize our present considerations by introducing the following definition of r -independence that may serve as a guidance in future constructions.

Definition For any r such that $2 \leq r \leq k$, by **r-independence**, we will understand the fact that for each $q \geq r$, all the q -dimensional joiners are equal to 1. \square

Thus, 2-independence is equivalent to the ordinary independence of the random variables X_1, \dots, X_k so, as usually, we have: $S(x_1, x_2, \dots, x_{k-1}, x_k) = S_1(x_1) S_2(x_2) \dots S_k(x_k)$.

It seems to be reasonable to assume that the underlying random variables X_1, \dots, X_k are only “pairwise dependent,” that is, to reduce the construction of a joint survival function $S(x_1, x_2, \dots, x_{k-1}, x_k)$ to construction of [not necessarily all] its bivariate marginals, say, $S_{ij}(x_i, x_j)$, where $1 \leq i < j \leq k$. Then we may apply scheme (7.3) by setting all the joiners having at least three arguments (and most of those having two arguments) to 1.

By the way, in this case formula (7.3) and some of its special realizations such as (7.8) should be a little redefined.

As a hint to this task realize that, in this case, instead of the arithmetic identity (7.1**), one may (preserving the main idea) apply the following, corresponding, arithmetic identity:

$$S(x_1, \dots, x_k) = \left[S(x_1, \dots, x_k) / \prod_{i < j} S_{ij}(x_i, x_j) \right] \prod_{i < j} S_{ij}(x_i, x_j). \quad (7.10)$$



Now, the corresponding to the former k -joiner, “bi-joiner” $J^{(2)}(x_1, \dots, x_k)$ is defined as

$$J^{(2)}(x_1, \dots, x_k) = S(x_1, \dots, x_k) / \prod_{i < j} S_{ij}(x_i, x_j),$$

while formula (7.3) is replaced by the following scheme:

$$S(x_1, \dots, x_k) = J^{(2)}(x_1, \dots, x_k) \prod_{i < j} S_{ij}(x_i, x_j). \quad (7.11)$$

Again, in concrete applications, most of the factors $S_{ij}(x_i, x_j)$ in (7.11) are reduced to products $S_i(x_i) S_j(x_j)$, which means the associated bi-joiners are reduced to 1.

As an important special case, one may consider the situation when we have

$$J^{(2)}(x_1, \dots, x_k) = 1.$$

Then (7.11) reduces to:

$$S(x_1, \dots, x_k) = \prod_{i < j} S_{ij}(x_i, x_j) \quad (7.12)$$

and now the theory reduces to the theory of bivariate probability distributions as described in [5] and [7].

This case is especially convincing when the only non-trivial factors in the product $\prod_{i < j} S_{ij}(x_i, x_j)$ are those for which any two element subsets $\{x_i, x_j\}$ of the variables set $\{x_1, \dots, x_k\}$ are disjoint.

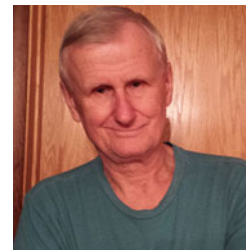
8. Freund, J.E.: A bivariate extension of the exponential distribution. *J. Am. Stat. Assoc.* **56**, 971–977 (1961)
9. Gumbel, E.J.: Bivariate exponential distributions. *J. Am. Stat. Assoc.* **55**, 698–707 (1960)
10. Kotz, S., Balakrishnan, N., Johnson, N.L.: *Continuous Multivariate Distributions*, Volume 1, Second edn. Wiley, New York (2000)
11. Marshall, A.W., Olkin, I.: A generalized bivariate exponential distribution. *J. Appl. Probab.* **4**, 291–303 (1967)
12. Sklar, A.: Fonctions de repartition a n dimensions et leurs marges. *Publications de l’Institut de Statistique de l’Universite de Paris.* **8**, 229–231 (1959)



Lidia Z. Filus is a Professor and Chair of Mathematics at Northeastern Illinois University in Chicago, Illinois, USA. She received her M.S. and Ph.D. in Mathematics from the University of Warsaw, Poland. She was awarded a research fellowship from the Center for Operations Research and Econometrics at the University of Louvain in Belgium, and from the University of Twente in the Netherlands. During her professional career she worked at the Warsaw School of Economics, the Polish Academy of Sciences, the University of Kansas, and Northeastern Illinois University in Chicago.

References

1. Aalen, O.O.: A linear regression model for the analysis of the life times. *Stat. Med.* **8**, 907–925 (1989)
2. Barry Arnold, ‘Private Communication’, 2017
3. Cox, D.R.: Regression models and life tables (with discussion). *J R Stat Soc, B.* **74**, 187–220 (1972)
4. Filus, J.K., Filus, L.Z.: A method for multivariate probability distributions construction via parameter dependence. *Commun. Stat. Theory Methods.* **42**(4), 716–721 (2013)
5. Filus, J.K., Filus, L.Z.: The Cox-Aalen models as framework for construction of bivariate probability distributions, universal representation. *J. Stat. Sci. Appl.* **5**, 56–63 (2017)
6. Filus, J.K., Filus, L.Z.: General method for construction of bivariate stochastic processes given two marginal processes. Presentation at 7-th International Conference on Risk Analysis, ICRA 7, Northeastern Illinois University, Chicago, 4 May 2017
7. Filus, J.K., Filus, L.Z.: A general (universal) form of multivariate survival functions in theoretical and modeling aspect of multicomponent system reliability analysis. In: Ram, M., Pham, H. (eds.) *Advances in Reliability Analysis and Its Applications* Springer Series in Reliability Engineering, pp. 319–342. Springer, Cham (2020). (online Dec. 12, 2019)



Jerzy K. Filus received his M.S. degree in Mathematics from the University of Warsaw, Poland, and his Ph.D. from the Systems Research Institute of the Polish Academy of Sciences. He was associated, among others, with the System Research Institute of the Polish Academy of Science, the Center for Operations Research and Econometrics, Belgium, the University of Twente, The Netherlands, and the Illinois Institute of Technology, Chicago, USA. Currently he is a retired Lecturer from the Dept. of Mathematics and Computer Sciences, Oakton College, Des Plaines, Illinois, USA. His research interests have mainly been focused on applied probability problems with emphasis on reliability modeling, and multivariate probability distributions and their extensions towards stochastic processes as well as fundamental concepts towards physics unification.



Hoang Pham

Contents

8.1	Introduction	137
8.2	Markov Processes	137
8.2.1	System Mean Time Between Failures.....	142
8.3	Counting Processes	145
8.3.1	Poisson Processes.....	145
8.3.2	Renewal Processes.....	146
8.3.3	Quasi-Renewal Processes.....	147
8.3.4	Nonhomogeneous Poisson Processes.....	148
8.4	Further Reading	149
Appendix A: Laplace Transformation Functions		149
References		151

Abstract

This chapter briefly discusses stochastic processes, including Markov processes, Poisson processes, renewal processes, quasi-renewal processes, and nonhomogeneous Poisson processes. The chapter also provides a short list of books for readers who are interested in advanced topics in stochastic processes.

Keywords

Markov processes · Poisson processes · Renewal processes · Stochastic processes · Nonhomogeneous Poisson processes

H. Pham (✉)
 Department of Industrial and Systems Engineering, Rutgers University, Piscataway, NJ, USA
 e-mail: hopham@soe.rutgers.edu

8.1 Introduction

Stochastic processes are used to describe the operation of a system over time. There are two main types of stochastic processes: continuous and discrete. A complex continuous process is a process describing a system transition from state to state. The simplest process that will be discussed here is a Markov process. In this case, the future behavior of the process does not depend on its past or present behavior. In many systems that arise in practice, however, past and present states of the system influence the future states, even if they do not uniquely determine them.

8.2 Markov Processes

In this section, we will discuss discrete stochastic processes. As an introduction to the Markov process, let us examine the following example.

Example 8.1 Consider a parallel system consisting of two components (see Fig. 8.1). From a reliability point of view, the states of the system can be described by

State 1:	Full operation (both components operating);
State 2:	One component is operating and one component has failed;
State 3:	Both components have failed.

Define

$$\begin{aligned}
 P_i(t) &= P[X(t) = i] \\
 &= P[\text{system is in state } i \text{ at time } t]
 \end{aligned}$$

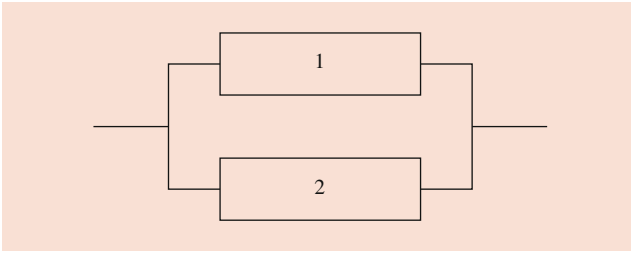


Fig. 8.1 A two-component parallel system

and

$$\begin{aligned} P_i(t + dt) &= P[X(t + dt) = i] \\ &= P[\text{system is in state } i \text{ at time } t + dt]. \end{aligned}$$

Define a random variable $X(t)$ which can assume the values 1, 2, or 3 corresponding to the states mentioned above. Since $X(t)$ is a random variable, one can discuss $P[X(t) = 1]$, $P[X(t) = 2]$ or the conditional probability $P[X(t_1) = 2 | X(t_0) = 1]$. Again, $X(t)$ is defined as a function of time t , while the conditional probability $P[X(t_1) = 2 | X(t_0) = 1]$ can be interpreted as the probability of being in state 2 at time t_1 , given that the system was in state 1 at time t_0 . In this example, the “state space” is discrete, i.e., 1, 2, 3, etc., and the parameter space (time) is continuous. The simple process described above is called a stochastic process: a process that develops over time (or space) in accordance with some probabilistic (stochastic) laws. There are many types of stochastic processes.

Here we emphasize the Markov process, which is a special type of stochastic process. Let the system be observed at discrete moments of time t_n , where $n = 0, 1, 2, \dots$, and let $X(t_n)$ denote the state of the system at time t_n .

Definition 8.1 Let $t_0 < t_1 < \dots < t_n$. If

$$\begin{aligned} &P[X(t_n) = x_n | X(t_{n-1}) \\ &= x_{n-1}, X(t_{n-2}) = x_{n-2}, \dots, X(t_0) = x_0] \\ &= P[X(t_n) = x_n | X(t_{n-1}) = x_{n-1}] \end{aligned} \quad (8.1)$$

then the process is called a *Markov process*.

From the definition of a Markov process, given the present state of the process, its behavior in the future does not depend on its behavior in the past.

The essential characteristic of a Markov process is that it is a process that has no memory; its future is determined by the present and not the past. If, in addition to having no memory, the process is such that it depends only on the difference $(t + dt) - t = dt$ and not the value of t – in other words $P[X(t + dt) = j | X(t) = i]$ is independent of t – then the process is Markov with stationary transition

probabilities or is homogeneous in time. This is the same property noted in exponential event times; in fact, referring back to the graphical representation of $X(t)$, the times between state changes are exponential if the process has stationary transition probabilities.

Thus, a Markov process which is homogeneous in time can describe processes with exponential event occurrence times. The random variable of the process is $X(t)$, the state variable rather than the time to failure used in the exponential failure density. To illustrate the types of processes that can be described, we now review the exponential distribution and its properties. Recall that, if X_1, X_2, \dots, X_n , are independent random variables, each with exponential density and a mean of $1/\lambda_i$, then $\min\{X_1, X_2, \dots, X_n\}$ has an exponential density with a mean of $(\sum \lambda_i)^{-1}$.

The significance of this property is as follows:

1. The failure behavior of components operated simultaneously can be characterized by an exponential density with a mean equal to the reciprocal of the sum of the failure rates.
2. The joint failure/repair behavior of a system where components are operating and/or undergoing repair can be characterized by an exponential density with a mean equal to the reciprocal of the sum of the failure and repair rates.
3. The failure/repair behavior of a system similar to that described in (2) above but further complicated by active and dormant operating states and sensing and switching can be characterized by an exponential density.

The above property means that almost all reliability and availability models can be characterized by a time-homogeneous Markov process if the various failure times and repair times are exponential. The notation for the Markov process is $\{X(t), t > 0\}$, where $X(t)$ is discrete (state space) and t is continuous (parameter space). By convention, this type of Markov process is called a continuous-parameter Markov chain.

From a reliability/availability viewpoint, there are two types of Markov processes. These are defined as follows:

1. *Absorbing process*: Contains an “absorbing state,” which is a state that, once entered, the system can never leave (e.g., a failure which aborts a flight or a mission).
2. *Ergodic process*: Contains no absorbing states, meaning that $X(t)$ can move around indefinitely (e.g., the operation of a ground power plant where failure only temporarily disrupts the operation).

Figure 8.2 presents a summary of Markov processes broken down into absorbing and ergodic categories. Both the reliability and the availability can be described in terms of the probability of the process or system being in defined “up” states, e.g., states 1 and 2 in the initial example. Likewise,

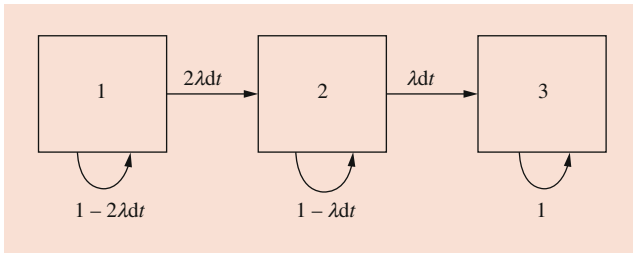


Fig. 8.2 State transition diagram for a two-component system

the mean time between failures (MTBF) can be described as the total time spent in the “up” states before proceeding to the absorbing state or failure state.

Define the incremental transition probability as

$$P_{ij}(dt) = P[X(t + dt) = j | X(t) = i].$$

This is the probability that the process [random variable $X(t)$] will move to state j during the increment t to $(t + dt)$, given that it was in state i at time t . Since we are dealing with time-homogeneous Markov processes (exponential failure and repair times), the incremental transition probabilities can be derived from an analysis of the exponential hazard function. It was shown that the hazard function for an exponential with a mean of $1/\lambda$ was just λ . This means that the limiting (as $dt \rightarrow 0$) conditional probability of an event occurring between t and $t + dt$, given that an event had not occurred at time t , is simply λ , in other words:

$$h(t) = \lim_{dt \rightarrow 0} \frac{P[t < X < t + dt | X > t]}{dt} = \lambda.$$

The equivalent statement for the random variable $X(t)$ is

$$h(t) dt = P[X(t + dt) = j | X(t) = i] = \lambda dt.$$

Now, $h(t) dt$ is in fact the incremental transition probability, so $P_{ij}(dt)$ can be stated in terms of the basic failure and/or repair rates. Define

$P_i(t)$: the probability that the system is in state i at time t

$r_{ij}(t)$: transition rate from state i to state j

In general, the differential equations can be written as follows:

$$\frac{\partial P_i(t)}{\partial t} = - \sum_j r_{ij}(t) P_i(t) + \sum_j r_{ji}(t) P_j(t). \quad (8.2)$$

Solving the above differential equations, one can obtain the time-dependent probability of each state.

Returning to Example 8.1, it is easy to construct a state transition showing the incremental transition probabilities between all possible states for the process:

State 1:	Both components operating
State 2:	One component up and one component down
State 3:	Both components down (absorbing state)

The loops in Fig. 8.2 indicate the probability of remaining in the present state during the dt increment

$$\begin{aligned} P_{11}(dt) &= 1 - 2\lambda dt & P_{12}(dt) &= 2\lambda dt \\ P_{21}(dt) &= 0 & P_{22}(dt) &= 1 - \lambda dt \\ P_{31}(dt) &= 0 & P_{32}(dt) &= 0 \\ P_{13}(dt) &= 0 \\ P_{23}(dt) &= \lambda dt \\ P_{33}(dt) &= 1 \end{aligned}$$

The zeros for P_{ij} , $i > j$ show that the process cannot go backwards: this is not a repair process. The zero on P_{13} shows that, for a process of this type, the probability of more than one event (e.g., failure, repair, etc.) occurring in the incremental time period dt approaches zero as dt approaches zero.

Except for the initial conditions of the process (the state in which the process starts), the process is completely specified by incremental transition probabilities. The reason that this is useful is that assuming exponential event (failure or repair) times allows us to characterize the process at any time t , since the process depends only on what happens between t and $(t + dt)$. The incremental transition probabilities can be arranged into a matrix in a way that depicts all possible statewide movements. Thus, for parallel configurations,

$$[P_{ij}(dt)] = \begin{pmatrix} 1 - 2\lambda dt & 2\lambda dt & 0 \\ 0 & 1 - \lambda dt & \lambda dt \\ 0 & 0 & 1 \end{pmatrix}$$

for $i, j = 1, 2$, or 3 . The matrix $[P_{ij}(dt)]$ is called the incremental, one-step transition matrix. It is a stochastic matrix (the rows sum to 1.0). As mentioned earlier, this matrix, along with the initial conditions, completely describes the process.

Now, $[P_{ij}(dt)]$ gives the probabilities of remaining or moving to all of the various states during the interval t to $t + dt$; hence,

$$\begin{aligned} P_1(t + dt) &= (1 - 2\lambda dt) P_1(t) \\ P_2(t + dt) &= 2\lambda dt P_1(t) + (1 - \lambda dt) P_2(t) \\ P_3(t + dt) &= \lambda dt P_2(t) + P_3(t) \end{aligned}$$

By algebraic manipulation, we have

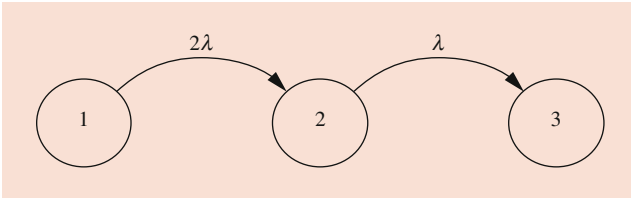


Fig. 8.3 Markov transition rate diagram for a two-component parallel system

$$\begin{aligned}\frac{[P_1(t+dt) - P_1(t)]}{dt} &= -2\lambda P_1(t), \\ \frac{[P_2(t+dt) - P_2(t)]}{dt} &= 2\lambda P_1(t) - \lambda P_2(t), \\ \frac{[P_3(t+dt) - P_3(t)]}{dt} &= \lambda P_2(t).\end{aligned}$$

Taking limits of both sides as $dt \rightarrow 0$, we obtain (also see Fig. 8.3):

$$\begin{aligned}P_1'(t) &= -2\lambda P_1(t), \\ P_2'(t) &= 2\lambda P_1(t) - \lambda P_2(t), \\ P_3'(t) &= \lambda P_2(t).\end{aligned}\quad (8.3)$$

The above system of linear first-order differential equations can be easily solved for $P_1(t)$ and $P_2(t)$, meaning that the reliability of the configuration can be obtained:

$$R(t) = \sum_{i=1}^2 P_i(t).\quad (8.4)$$

Actually, there is no need to solve all three equations, only the first two, because $P_3(t)$ does not appear and also $P_3(t) = [1 - P_1(t)] - P_2(t)$. The system of linear, first-order differential equations can be solved by various means, including both manual and machine methods. We use manual methods employing the Laplace transform (see Appendix A) here.

$$\begin{aligned}L[P_i(t)] &= \int_0^\infty e^{-st} P_i(t) dt = f_i(s), \\ L[P_i'(t)] &= \int_0^\infty e^{-st} P_i'(t) dt = s f_i(s) - P_i(0).\end{aligned}\quad (8.5)$$

Application of the Laplace transform will allow us to transform the system of linear, first-order differential equations into a system of linear algebraic equations that can easily be solved, and solutions of $P_i(t)$ can be determined via the inverse transforms.

Returning to the example, the initial condition of a parallel configuration is assumed to be “fully up”, such that

$$P_1(t=0) = 1, P_2(t=0) = 0, P_3(t=0) = 0.$$

Transforming the equations for $P_1'(t)$ and $P_2'(t)$ gives

$$\begin{aligned}s f_1(s) - P_1(t)|_{t=0} &= -2\lambda f_1(s), \\ s f_2(s) - P_2(t)|_{t=0} &= 2\lambda f_1(s) - \lambda f_2(s).\end{aligned}$$

Evaluating $P_1(t)$ and $P_2(t)$ at $t = 0$ gives

$$\begin{aligned}s f_1(s) - 1 &= -2\lambda f_1(s), \\ s f_2(s) - 0 &= 2\lambda f_1(s) - \lambda f_2(s).\end{aligned}$$

from which we obtain

$$\begin{aligned}(s + 2\lambda) f_1(s) &= 1, \\ -2\lambda f_1(s) + (s + \lambda) f_2(s) &= 0.\end{aligned}$$

Solving the above equations for $f_1(s)$ and $f_2(s)$, we have

$$\begin{aligned}f_1(s) &= \frac{1}{(s + 2\lambda)}, \\ f_2(s) &= \frac{2\lambda}{[(s + 2\lambda)(s + \lambda)]}.\end{aligned}$$

From the inverse Laplace transforms in Appendix A,

$$\begin{aligned}P_1(t) &= e^{-2\lambda t}, \\ P_2(t) &= 2e^{-\lambda t} - 2e^{-2\lambda t}, \\ R(t) = P_1(t) + P_2(t) &= 2e^{-\lambda t} - e^{-2\lambda t}.\end{aligned}\quad (8.6)$$

The example given above is that of a simple absorbing process where we are concerned about reliability. If a repair capability were added to the model in the form of a repair rate μ , the methodology would remain the same, with only the final result changing. With a repair rate μ added to the parallel configuration (see Fig. 8.4), the incremental transition matrix would be

$$[P_{ij}(dt)] = \begin{pmatrix} 1 - 2\lambda dt & 2\lambda dt & 0 \\ \mu dt & 1 - (\lambda + \mu) dt & \lambda dt \\ 0 & 0 & 1 \end{pmatrix}.$$

The differential equations would become (see Fig. 8.4)

$$\begin{aligned}P_1'(t) &= -2\lambda P_1(t) + \mu P_2(t), \\ P_2'(t) &= 2\lambda P_1(t) + (\lambda + \mu) P_2(t),\end{aligned}$$

and the transformed equations would become

$$\begin{aligned}(s + 2\lambda) f_1(s) - \mu f_2(s) &= 1, \\ -2\lambda f_1(s) + (s + \lambda + \mu) f_2(s) &= 0.\end{aligned}$$

Hence, we obtain

$$\begin{aligned}f_1(s) &= \frac{(s + \lambda + \mu)}{(s - s_1)(s - s_2)}, \\ f_2(s) &= \frac{2\lambda}{(s - s_1)(s - s_2)},\end{aligned}\quad (8.7)$$

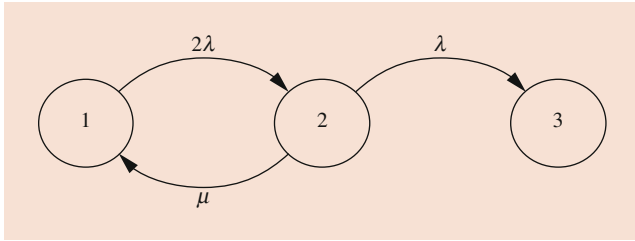


Fig. 8.4 Markov transition rate diagram for a two-component parallel repairable system

where

$$s_1 = \frac{-(3\lambda + \mu) + \sqrt{(3\lambda + \mu)^2 - 8\lambda^2}}{2}, \quad (8.8)$$

$$s_2 = \frac{-(3\lambda + \mu) - \sqrt{(3\lambda + \mu)^2 - 8\lambda^2}}{2}.$$

Using the Laplace transform, we obtain

$$P_1(t) = \frac{(s_1 + \lambda + \mu) e^{-s_1 t}}{(s_1 - s_2)} + \frac{(s_2 + \lambda + \mu) e^{-s_2 t}}{(s_2 - s_1)}, \quad (8.9)$$

$$P_2(t) = \frac{2\lambda e^{-s_1 t}}{(s_1 - s_2)} + \frac{2\lambda e^{-s_2 t}}{(s_2 - s_1)},$$

where s_1 and s_2 are given in Eq. (8.8).

Thus, the reliability of a two-component parallel repairable system is given by

$$R(t) = P_1(t) + P_2(t) = \frac{(s_1 + 3\lambda + \mu) e^{-s_1 t} - (s_2 + 3\lambda + \mu) e^{-s_2 t}}{(s_1 - s_2)} \quad (8.10)$$

Example 8.2 Consider a three-unit shared load parallel system where

- λ_0 is the constant failure rate of a unit when all the three units are operational;
- λ_h is the constant failure rate of each of the two surviving units, each of which shares half of the total load; and
- λ_f is the constant failure rate of a unit at full load.

For a shared-load parallel system to fail, all the units in the system must fail. We now derive the reliability of a three-unit shared-load parallel system using the Markov model.

In reliability analysis, for the three-unit load-sharing system to work the following events would be considered:

- Event 1:* All the three units are working until the end of the mission time t where each unit shares one-third of the total load.
- Event 2:* All the three units are working until time t_1 (each shares one-third of the total load). At time t_1 , one of the units (say unit 1) fails, and the other two units (say units 2

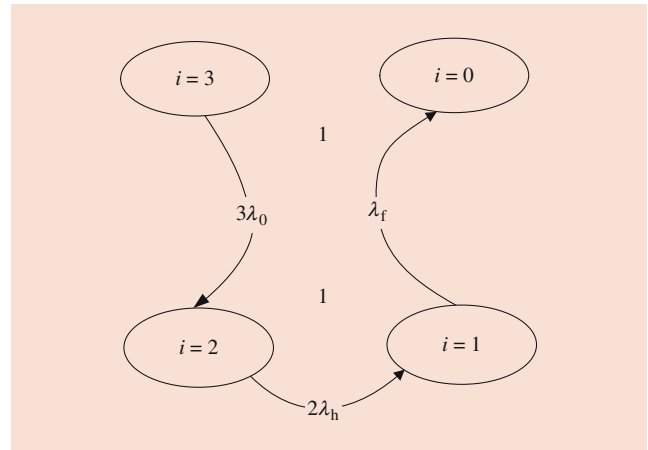


Fig. 8.5 Markov model diagram for a three-unit shared-load parallel system

and 3) remain to work until the mission time t . Here, once a unit fails at time t_1 , the remaining two working units would take half each of the total load and have a constant rate λ_h . As for all identical units, there are three possibilities under this situation.

Event 3: All the three units are working until time t_1 (each shares one-third of the total load) when one (say unit 1) of the three units fails. At time t_2 , ($t_2 > t_1$) one more unit fails (say unit 2) and the remaining unit works until the end of the mission time t . Under this event, there are six possibilities that the probability of two units failing before time t and only one unit remains to work until time t .

Define state i represents that i components are working. Let $P_i(t)$ denote the probability that the system is in state i at time t for $i = 0, 1, 2, 3$. Figure 8.5 below shows the Markov diagram of the system.

The Markov modeling system of differential equations based on Fig. 8.1 can be easily derived as follows:

$$\begin{cases} \frac{dP_3(t)}{dt} = -3\lambda_0 P_3(t) \\ \frac{dP_2(t)}{dt} = 3\lambda_0 P_3(t) - 2\lambda_h P_2(t) \\ \frac{dP_1(t)}{dt} = 2\lambda_h P_2(t) - \lambda_f P_1(t) \\ \frac{dP_0(t)}{dt} = \lambda_f P_1(t) \\ P_3(0) = 1 \\ P_j(0) = 0, j \neq 3 \\ P_0(t) + P_1(t) + P_2(t) + P_3(t) = 1 \end{cases} \quad (8.11)$$

Solving the above differential equations using the Laplace transform method (see Appendix A), we can easily obtain the following results:

$$P_3(t) = e^{-3\lambda_0 t}$$

$$P_2(t) = \frac{3\lambda_0}{3\lambda_h - 3\lambda_0} (e^{-3\lambda_0 t} - e^{-2\lambda_h t})$$

$$P_1(t) = \frac{6\lambda_0\lambda_h}{(2\lambda_h - 3\lambda_0)} \left[\frac{e^{-3\lambda_0 t}}{(\lambda_f - 3\lambda_0)} - \frac{e^{-2\lambda_h t}}{(\lambda_f - 2\lambda_h)} + \frac{(2\lambda_h - 3\lambda_0)e^{-\lambda_f t}}{(\lambda_f - 3\lambda_0)(\lambda_f - 2\lambda_h)} \right] \quad (8.12)$$

Hence, the reliability of a three-unit shared load parallel system can be obtained as follows:

$$\begin{aligned} R(t) &= P_3(t) + P_2(t) + P_1(t) \\ &= e^{-3\lambda_0 t} + \frac{3\lambda_0}{2\lambda_h - 3\lambda_0} (e^{-3\lambda_0 t} - e^{-2\lambda_h t}) \\ &\quad + \frac{6\lambda_0\lambda_h}{(2\lambda_h - 3\lambda_0)} \left[\frac{e^{-3\lambda_0 t}}{(\lambda_f - 3\lambda_0)} - \frac{e^{-2\lambda_h t}}{(\lambda_f - 2\lambda_h)} \right. \\ &\quad \left. + \frac{(2\lambda_h - 3\lambda_0)e^{-\lambda_f t}}{(\lambda_f - 3\lambda_0)(\lambda_f - 2\lambda_h)} \right] \end{aligned} \quad (8.13)$$

8.2.1 System Mean Time Between Failures

Another parameter of interest for absorbing Markov processes is the MTBF. Recalling Example 8.1 of a parallel configuration with repair, the differential equations $P_1'(t)$ and $P_2'(t)$ describing the process were

$$\begin{aligned} P_1'(t) &= -2\lambda P_1(t) + \mu P_2(t), \\ P_2'(t) &= 2\lambda P_1(t) - (\lambda + \mu) P_2(t). \end{aligned}$$

Integrating both sides of the above equations yields

$$\begin{aligned} \int_0^\infty P_1'(t) dt &= -2\lambda \int_0^\infty P_1(t) dt + \mu \int_0^\infty P_2(t) dt, \\ \int_0^\infty P_2'(t) dt &= 2\lambda \int_0^\infty P_1(t) dt - (\lambda + \mu) \int_0^\infty P_2(t) dt. \end{aligned}$$

For the repairable system, we have

$$\int_0^\infty R(t) dt = \text{MTBF}. \quad (8.14)$$

Similarly,

$$\begin{aligned} \int_0^\infty P_1(t) dt &= \text{mean time spent in state 1, and} \\ \int_0^\infty P_2(t) dt &= \text{mean time spent in state 2.} \end{aligned}$$

Designating these mean times as T_1 and T_2 , respectively, we have

$$\begin{aligned} P_1(t) dt \Big|_0^\infty &= -2\lambda T_1 + \mu T_2, \\ P_2(t) dt \Big|_0^\infty &= 2\lambda T_1 - (\lambda + \mu) T_2. \end{aligned}$$

But $P_1(t) = 0$ as $t \rightarrow \infty$ and $P_1(t) = 1$ for $t = 0$. Likewise, $P_2(t) = 0$ as $t \rightarrow \infty$ and $P_2(t) = 0$ for $t = 0$. Thus,

$$\begin{aligned} -1 &= -2\lambda T_1 + \mu T_2, \\ 0 &= 2\lambda T_1 - (\lambda + \mu) T_2, \end{aligned}$$

or, equivalently,

$$\begin{pmatrix} -1 \\ 0 \end{pmatrix} = \begin{pmatrix} -2\lambda & \mu \\ 2\lambda & -(\lambda + \mu) \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix}.$$

Therefore,

$$\begin{aligned} T_1 &= \frac{(\lambda + \mu)}{2\lambda^2}, \quad T_2 = \frac{1}{\lambda}, \\ \text{MTBF} &= T_1 + T_2 = \frac{(\lambda + \mu)}{2\lambda^2} + \frac{1}{\lambda} = \frac{(3\lambda + \mu)}{2\lambda^2}. \end{aligned} \quad (8.15)$$

The MTBF for unmaintained processes is developed in exactly the same way as just shown.

The last case to consider for absorbing processes is that of the availability of a maintained system. The difference between reliability and availability is somewhat subtle for absorbing processes. A good example is that of a communications system where the mission would continue if such a system failed temporarily, but if it failed permanently the mission would be aborted. Consider a cold-standby system consisting of two units: one main unit and one spare unit [1]:

State 1:	Main unit operating and the spare is OK
State 2:	Main unit out and restoration underway
State 3:	Spare unit is installed and operating
State 4:	Permanent failure (no spare available)

The incremental transition matrix is given by

$$[P_{ij}(dt)] = \begin{pmatrix} 1 - \lambda dt & \lambda dt & 0 & 0 \\ 0 & 1 - \mu dt & \mu dt & 0 \\ 0 & 0 & 1 - \lambda dt & \lambda dt \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

We obtain

$$\begin{aligned} P_1'(t) &= -\lambda P_1(t), \\ P_2'(t) &= \lambda P_1(t) - \mu P_2(t), \\ P_3'(t) &= \mu P_2(t) - \lambda P_3(t). \end{aligned}$$

Using the Laplace transform, we obtain the following results.

The probability of full-up performance $P_1(t)$ is given by

$$P_1(t) = e^{-\lambda t}.$$

The probability of a down system that is under repair $P_2(t)$ is

$$P_2(t) = \left(\frac{\lambda}{\lambda - \mu} \right) (e^{-\mu t} - e^{-\lambda t}).$$

Similarly, the probability of a fully up system with no spare available $P_3(t)$ is

$$P_3(t) = \left(\frac{\lambda\mu}{(\lambda - \mu)^2} \right) [e^{-\mu t} - e^{-\lambda t} - (\lambda - \mu)t e^{-\lambda t}].$$

Hence, the point availability $A(t)$ is given by

$$A(t) = P_1(t) + P_3(t). \tag{8.16}$$

If average or interval availability is required, this is achieved by

$$\left(\frac{1}{t} \right) \int_0^T A(t) dt = \left(\frac{1}{t} \right) \int_0^T [P_1(t) + P_3(t)] dt,$$

where T is the interval of concern.

Ergodic processes, as opposed to absorbing processes, do not have any absorbing states, and hence movement between states can go on indefinitely. For the latter reason, availability (point, steady-state, or interval) is the only meaningful measure. As an example of an ergodic process, we will use a ground-based power unit configured in parallel.

The parallel units are identical, each with exponential failure and repair times with means $1/\lambda$ and $1/\mu$, respectively. Assume a two-repairmen capability if required (both units down), then

State 1:	Fully up (both units operating)
State 2:	One unit down and under repair (other unit up)
State 3:	Both units down and under repair

It should be noted that, as in the case of failure events, two or more repairs cannot be made in the dt interval.

$$[P_{ij}(dt)] = \begin{pmatrix} 1 - 2\lambda dt & 2\lambda dt & 0 \\ \mu dt & 1 - (\lambda + \mu) dt & \lambda dt \\ 0 & 2\mu dt & 1 - 2\mu dt \end{pmatrix}.$$

Case I: *Point Availability – Ergodic Process.* For an ergodic process, as $t \rightarrow \infty$ the availability settles down to a constant level. Point availability allows us to study the process before this “settling down,” and it reflects the initial conditions in the process. We can obtain a solution for the point availability in a similar way to that for absorbing

processes, except that the last row and column of the transition matrix must be retained and entered into the system of equations. For example, the system of differential equations becomes

$$\begin{pmatrix} P_1'(t) \\ P_2'(t) \\ P_3'(t) \end{pmatrix} = \begin{pmatrix} -2\lambda & \mu & 0 \\ 2\lambda & -(\lambda + \mu) & 2\mu \\ 0 & \lambda & -2\mu \end{pmatrix} \begin{pmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \end{pmatrix}.$$

Similar to the absorbing case, the Laplace transform can be used to solve for $P_1(t)$, $P_2(t)$ and $P_3(t)$; the point availability $A(t)$ is given by

$$A(t) = P_1(t) + P_2(t).$$

Case II: *Interval Availability – Ergodic Process.* This is the same as the absorbing case, with integration over the time period T of interest. The interval availability, $A(T)$, is

$$A(T) = \frac{1}{T} \int_0^T A(t) dt. \tag{8.17}$$

Case III: *Steady State Availability – Ergodic Process.* Here, the process is examined as $t \rightarrow \infty$, with complete “washout” of the initial conditions. By letting $t \rightarrow \infty$, the system of differential equations can be transformed into linear algebraic equations. Thus,

$$\begin{aligned} & \lim_{t \rightarrow \infty} \begin{pmatrix} P_1'(t) \\ P_2'(t) \\ P_3'(t) \end{pmatrix} \\ &= \lim_{t \rightarrow \infty} \begin{pmatrix} -2\lambda & \mu & 0 \\ 2\lambda & -(\lambda + \mu) & 2\mu \\ 0 & \lambda & -2\mu \end{pmatrix} \begin{pmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \end{pmatrix}. \end{aligned}$$

As $t \rightarrow \infty$, $P_i(t) \rightarrow \text{constant}$ and $P_i'(t) \rightarrow 0$. This leads to an unsolvable system, namely,

$$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -2\lambda & \mu & 0 \\ 2\lambda & -(\lambda + \mu) & 2\mu \\ 0 & \lambda & -2\mu \end{pmatrix} \begin{pmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \end{pmatrix}.$$

To avoid the above difficulty, an additional equation is introduced:

$$\sum_{i=1}^3 P_i(t) = 1.$$

With the introduction of the new equation, one of the original equations is deleted and a new system is formed:

$$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ -2\lambda & \mu & 0 \\ 2\lambda & -(\lambda + \mu) & 2\mu \end{pmatrix} \begin{pmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \end{pmatrix}$$

or, equivalently,

$$\begin{pmatrix} P_1(t) \\ P_2(t) \\ P_3(t) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ -2\lambda & \mu & 0 \\ 2\lambda & -(\lambda + \mu) & 2\mu \end{pmatrix}^{-1} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}.$$

We now obtain the following results:

$$P_1(t) = \frac{\mu^2}{(\mu + \lambda)^2},$$

$$P_2(t) = \frac{2\lambda\mu}{(\mu + \lambda)^2},$$

and

$$P_3(t) = 1 - P_1(t) - P_2(t)$$

$$= \frac{\lambda^2}{(\mu + \lambda)^2}.$$

Therefore, the steady state availability $A(\infty)$ is given by

$$A_3(\infty) = P_1(t) + P_2(t)$$

$$= \frac{\mu(\mu + 2\lambda)}{(\mu + \lambda)^2}. \quad (8.18)$$

Note that Markov methods can also be employed when failure or repair times are not exponential but can be represented as the sum of exponential times with identical means (an Erlang distribution or gamma distribution with integer-valued shape parameters). Basically, the method involves introducing “dummy” states which, although being of no particular interest in themselves, change the hazard function from constant to increasing.

Example 8.3 A system is composed of eight identical active power supplies, at least seven of the eight are required for the system to function. In other words, when two of the eight power supplies fail, the system fails. When all eight power supplies are operating, each has a constant failure rate λ_a per hour. If one power supply fails, each remaining power supply has a failure rate λ_b per hour where $\lambda_a \leq \lambda_b$. We assume that a failed power supply can be repaired with a constant rate μ per hour. The system reliability function, $R(t)$, is defined as the probability that the system continues to function throughout the interval $(0, t)$. Here we wish to determine the system mean time to failure (MTTF).

Define

State 0: All 8 units are working

State 1: 7 units are working

State 2: More than one unit failed and system does not work

The initial condition: $P_0(0) = 1, P_1(0) = P_2(0) = 0$

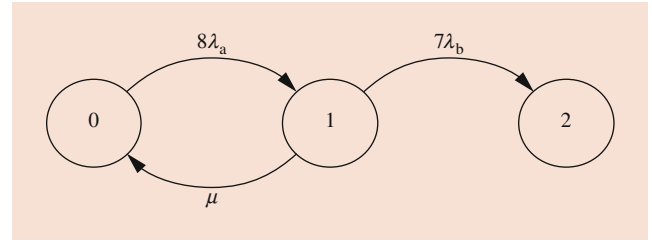


Fig. 8.6 Markov transition rate diagram for a 7-out-of-8 dependent system

The Markov modeling of differential equations (see Fig. 8.6) can be written as follows:

$$P'_0(t) = -8\lambda_a P_0(t) + \mu P_1(t)$$

$$P'_1(t) = 8\lambda_a P_0(t) - (7\lambda_b + \mu) P_1(t)$$

$$P'_2(t) = 7\lambda_b P_1(t)$$

Using the Laplace transform, we obtain

$$\begin{cases} sF_0(s) - P_0(0) = -8\lambda_a F_0(s) + \mu F_1(s) \\ sF_1(s) - P_1(0) = 8\lambda_a F_0(s) - (7\lambda_b + \mu) F_1(s) \\ sF_2(s) - P_2(0) = 7\lambda_b F_1(s) \end{cases} \quad (8.19)$$

When $s = 0$:

$$F_i(0) = \int_0^\infty P_i(t) dt.$$

Thus, the system reliability function and system MTTF, respectively, are

$$R(t) = P_0(t) + P_1(t). \quad (8.20)$$

and

$$\text{MTTF} = \int_0^\infty R(t) dt = \int_0^\infty [P_0(t) + P_1(t)] dt = \sum_{i=1}^2 F_i(0). \quad (8.21)$$

From Eq. (8.19), when $s = 0$, we have

$$\begin{cases} -1 = -8\lambda_a F_0(0) + \mu F_1(0) \\ 0 = 8\lambda_a F_0(0) - (7\lambda_b + \mu) F_1(0). \end{cases} \quad (8.22)$$

From Eq. (8.22), after some arrangements, we can obtain

$$7\lambda_b F_1(0) = 1 \Rightarrow F_1(0) = \frac{1}{7\lambda_b}$$

and

$$\begin{aligned} F_0(0) &= \frac{7\lambda_b + \mu}{8\lambda_a} F_1(0) \\ &= \frac{7\lambda_b + \mu}{8\lambda_a} \frac{1}{7\lambda_b} = \frac{7\lambda_b + \mu}{56\lambda_a\lambda_b} \end{aligned}$$

From Eq. (8.21), the system MTTF can be obtained

$$\begin{aligned} \text{MTTF} &= \int_0^\infty R(t) dt = \int_0^\infty [P_0(t) + P_1(t)] dt \\ &= F_0(0) + F_1(0) \\ &= \frac{7\lambda_b + \mu}{56\lambda_a\lambda_b} + \frac{1}{7\lambda_b} = \frac{\mu + 8\lambda_a + 7\lambda_b}{56\lambda_a\lambda_b}. \end{aligned}$$

Given $\lambda_a = 3 \times 10^{-3} = 0.003$, $\lambda_b = 5 \times 10^{-2} = 0.05$, and $\mu = 0.8$, then the system mean time to failure is given by:

$$\begin{aligned} \text{MTTF} &= \frac{\mu + 8\lambda_a + 7\lambda_b}{56\lambda_a\lambda_b} \\ &= \frac{0.8 + 8(0.003) + 7(0.05)}{56(0.003)(0.05)} = \frac{1.174}{0.0084} = 139.762 \text{ hours.} \end{aligned}$$

8.3 Counting Processes

Among various discrete stochastic processes, counting processes are widely used in engineering statistics to describe the appearance of events in time, such as failures, the number of perfect repairs, etc. The simplest counting process is a Poisson process. The Poisson process plays a special role in many applications related to reliability [1]. A classic example of such an application is the decay of uranium. Here, radioactive particles from nuclear material strike a certain target in accordance with a Poisson process of some fixed intensity. One well-known counting process is the so-called renewal process. This process is described as a sequence of events where the intervals between the events are independent and identically distributed random variables. In reliability theory, this type of mathematical model is used to describe the number of occurrences of an event or the number of renewals (i.e., replacements of objects) over a time interval. A light bulb is shining in your living room and it blows up suddenly. You replace it by a new bulb. It lasts for a few months, then burns out again. You then replace it again, and so on. One would be interested to know about the total number of bulbs is needed to be replaced in 2 years.

In this subsection, we discuss the concepts and properties of the Poisson process, renewal process, quasi-renewal process, and nonhomogeneous Poisson process.

A non-negative, integer-valued stochastic process $N(t)$ is called a counting process if $N(t)$ represents the total number of occurrences of an event in the time interval $[0, t]$ and satisfies these two properties:

1. If $t_1 < t_2$, then $N(t_1) \leq N(t_2)$
2. If $t_1 < t_2$, then $N(t_2) - N(t_1)$ is the number of occurrences of the event in the interval $[t_1, t_2]$.

For example, if $N(t)$ equals the number of persons who have entered a restaurant at or prior to time t , then $N(t)$ is a counting process in which an event occurs whenever a person enters the restaurant.

8.3.1 Poisson Processes

One of the most important counting processes is the Poisson process.

Definition 8.3 A counting process $N(t)$ is said to be a Poisson process with intensity λ if

1. The failure process $N(t)$ has stationary independent increments
2. The number of failures in any time interval of length s has a Poisson distribution with a mean of λs ; in other words

$$P\{N(t+s) - N(t) = n\} = \frac{e^{-\lambda s} (\lambda s)^n}{n!} \quad (8.23)$$

$$n = 0, 1, 2, \dots;$$

3. The initial condition is $N(0) = 0$

This model is also called a homogeneous Poisson process, indicating that the failure rate λ does not depend on time t . In other words, the number of failures that occur during the time interval $(t, t+s)$ does not depend on the current time t , only the length of the time interval s . A counting process is said to possess independent increments if the number of events in disjoint time intervals are independent.

For a stochastic process with independent increments, the autocovariance function is

$$\begin{aligned} \text{Cov}[X(t_1), X(t_2)] &= \begin{cases} \text{Var}[N(t_1+s) - N(t_2)] & \text{for } 0 < t_2 - t_1 < s \\ 0 & \text{otherwise} \end{cases}, \end{aligned}$$

where

$$X(t) = N(t+s) - N(t).$$

If $X(t)$ is Poisson-distributed, then the variance of the Poisson distribution is

$$\begin{aligned} \text{Cov}[X(t_1), X(t_2)] &= \begin{cases} \lambda [s - (t_2 - t_1)] & \text{for } 0 < t_2 - t_1 < s \\ 0 & \text{otherwise} \end{cases}. \end{aligned}$$

This result shows that the Poisson increment process is covariance stationary. We now present several properties of the Poisson process.

Property 8.1 The sum of independent Poisson processes $N_1(t), N_2(t), \dots, N_k(t)$ with mean values $\lambda_1 t, \lambda_2 t, \dots, \lambda_k t$, respectively, is also a Poisson process with mean $\left(\sum_{i=1}^k \lambda_i\right)t$. In other words, the sum of the independent Poisson processes is also a Poisson process with a mean that is equal to the sum of the means of the individual Poisson processes.

Property 8.2 The difference between two independent Poisson processes, $N_1(t)$ and $N_2(t)$, with mean $\lambda_1 t$ and $\lambda_2 t$, respectively, is not a Poisson process. Instead, it has a probability mass function of

$$\begin{aligned} P[N_1(t) - N_2(t) = k] \\ = e^{-(\lambda_1 + \lambda_2)t} \left(\frac{\lambda_1}{\lambda_2}\right)^{\frac{k}{2}} I_k(2\sqrt{\lambda_1 \lambda_2} t), \end{aligned}$$

where $I_k(\cdot)$ is a modified Bessel function of order k .

Property 8.3 If the Poisson process $N(t)$ with mean λt is filtered such that not every occurrence of the event is counted, then the process has a constant probability p of being counted. The result of this process is a Poisson process with mean $\lambda p t$.

Property 8.4 Let $N(t)$ be a Poisson process and Y_i a family of independent and identically distributed random variables which are also independent of $N(t)$. A stochastic process $X(t)$ is said to be a compound Poisson process if it can be represented as

$$X(t) = \sum_{i=1}^{N(t)} Y_i.$$

8.3.2 Renewal Processes

A renewal process is a more general case of the Poisson process in which the inter-arrival times of the process or the times between failures do not necessarily follow the exponential distribution. For convenience, we will call the occurrence of an event a renewal, the inter-arrival time the renewal period, and the waiting time the renewal time.

Definition 8.3 A counting process $N(t)$ that represents the total number of occurrences of an event in the time interval $(0, t]$ is called a renewal process if the times between the failures are independent and identically distributed random variables.

The probability that exactly n failures occur by time t can be written as

$$P[N(t) = n] = P[N(t) \geq n] - P[N(t) > n]. \quad (8.24)$$

Note that the times between the failures are T_1, T_2, \dots, T_n , so the failures occurring at time W_k are

$$W_k = \sum_{i=1}^k T_i$$

and

$$T_k = W_k - W_{k-1}.$$

Thus,

$$\begin{aligned} P[N(t) = n] &= P[N(t) \geq n] - P[N(t) > n] \\ &= P[W_n \leq t] - P[W_{n+1} \leq t] \\ &= F_n(t) - F_{n+1}(t), \end{aligned} \quad (8.25)$$

where $F_n(t)$ is the cumulative distribution function for the time of the n th failure and $n = 0, 1, 2, \dots$

Example 8.4 Consider a software testing model for which the time at which an error is found during the testing phase has an exponential distribution with a failure rate of X . It can be shown that the time of the n th failure follows the gamma distribution with parameters k and n . From Eq. (8.24), we obtain

$$\begin{aligned} P[N(t) = n] &= P[N(t) \leq n] - P[N(t) \leq n-1] \\ &= \sum_{k=0}^n \frac{(\lambda t)^k}{k!} e^{-\lambda t} - \sum_{k=0}^{n-1} \frac{(\lambda t)^k}{k!} e^{-\lambda t} \\ &= \frac{(\lambda t)^n}{n!} e^{-\lambda t} \text{ for } n = 0, 1, 2, \dots \end{aligned} \quad (8.26)$$

Several important properties of the renewal function are given below.

Property 8.5 The mean value function of the renewal process, denoted by $m(t)$, is equal to the sum of the distribution functions for all renewal times, that is,

$$\begin{aligned} m(t) &= E[N(t)] \\ &= \sum_{n=1}^{\infty} F_n(t). \end{aligned} \quad (8.27)$$

Property 8.6 The renewal function $m(t)$ satisfies the following equation:

$$m(t) = F_a(t) + \int_0^t m(t-s) dF_a(s), \quad (8.28)$$

where $F_a(t)$ is the distribution function of the inter-arrival time or the renewal period.

In general, let $y(t)$ be an unknown function to be evaluated and $x(t)$ be any non-negative and integrable function associated with the renewal process. Assume that $F_a(t)$ is the distribution function of the renewal period. We can then obtain the following result.

Property 8.7 Let the renewal equation be

$$y(t) = x(t) + \int_0^t y(t-s) dF_a(s). \quad (8.29)$$

Then its solution is given by

$$y(t) = x(t) + \int_0^t x(t-s) dm(s)$$

where $m(t)$ is the mean value function of the renewal process.

The proof of the above property can be easily derived using the Laplace transform. Let $x(t) = a$. Thus, in Property 8.7, the solution $y(t)$ is given by

$$\begin{aligned} y(t) &= x(t) + \int_0^t x(t-s) dm(s) \\ &= a + \int_0^t a dm(s) \\ &= a \{1 + E[N(t)]\}. \end{aligned}$$

8.3.3 Quasi-Renewal Processes

In this section we discuss a general renewal process: the quasi-renewal process. Let $\{N(t), t > 0\}$ be a counting process and let X_n be the time between the $(n-1)$ th and the n th event of this process, $n \geq 1$.

Definition 8.4 [2] If the sequence of non-negative random variables $\{X_1, X_2, \dots\}$ is independent and

$$X_i = \alpha X_{i-1} \quad (8.30)$$

for $i \geq 2$ where $\alpha > 0$ is a constant, then the counting process $\{N(t), t \geq 0\}$ is said to be a quasi-renewal process with parameter α and the first inter-arrival time X_1 .

When $\alpha = 1$, this process becomes the ordinary renewal process. This quasi-renewal process can be used to model reliability growth processes in software testing phases and hardware burn-in stages for $\alpha > 1$, and in hardware maintenance processes when $\alpha \leq 1$.

Assume that the probability density function (pdf), cumulative distribution function (cdf), survival function, and failure rate of random variable X_1 are $f_1(x)$, $F_1(x)$, $s_1(x)$ and

$r_1(x)$, respectively. Then the pdf, cdf, survival function, and failure rate of X_n for $n = 1, 2, 3, \dots$ are, respectively, given below [2]:

$$\begin{aligned} f_n(x) &= \frac{1}{\alpha^{n-1}} f_1\left(\frac{1}{\alpha^{n-1}}x\right), \\ F_n(x) &= F_1\left(\frac{1}{\alpha^{n-1}}x\right), \\ s_n(x) &= s_1\left(\frac{1}{\alpha^{n-1}}x\right), \\ r_n(x) &= \frac{1}{\alpha^{n-1}} r_1\left(\frac{1}{\alpha^{n-1}}x\right). \end{aligned} \quad (8.31)$$

Similarly, the mean and variance of X_n is given as

$$\begin{aligned} E(X_n) &= \alpha^{n-1} E(X_1), \\ \text{Var}(X_n) &= \alpha^{2n-2} \text{Var}(X_1). \end{aligned} \quad (8.32)$$

Because of the non-negativity of X_1 , and the fact that X_1 is not identically 0, we obtain

$$E(X_1) = \mu_1 \neq 0.$$

It is worth noting that the shape parameters for X_n are the same for $n = 1, 2, 3, \dots$ for a quasi-renewal process if X_1 follows the gamma, Weibull, or log normal distribution.

This means that the shape parameters of the inter-arrival time will not change after “renewal”. In software reliability, the assumption that the software debugging process does not change the error-free distribution seems reasonable. Thus, if a quasi-renewal process model is used, the error-free times that occur during software debugging will have the same shape parameters. In this sense, a quasi-renewal process is suitable for modeling the increase in software reliability. It is worth noting that

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{E(X_1 + X_2 + \dots + X_n)}{n} &= \lim_{n \rightarrow \infty} \frac{\mu_1 (1 - \alpha^n)}{(1 - \alpha) n} \\ &= \begin{cases} 0 & \text{if } \alpha < 1, \\ \infty & \text{if } \alpha > 1. \end{cases} \end{aligned} \quad (8.33)$$

Therefore, if the inter-arrival time represents the error-free time of a software system, then the average error-free time approaches infinity when its debugging process has been operating for a long debugging time.

Distribution of $N(t)$

Consider a quasi-renewal process with parameter α and a first inter-arrival time X_1 . Clearly, the total number of renewals $N(t)$ that occur up to time t has the following relationship to the arrival time of the n th renewal SS_n :

$$N(t) \geq n \quad \text{if and only if} \quad SS_n \leq t.$$

In other words, $N(t)$ is at least n if and only if the n th renewal occurs prior to time t . It is easily seen that

$$SS_n = \sum_{i=1}^n X_i = \sum_{i=1}^n \alpha^{i-1} X_1 \quad \text{for } n \geq 1. \quad (8.34)$$

Here, $SS_0 = 0$. Thus, we have

$$\begin{aligned} P\{N(t) = n\} &= P\{N(t) \geq n\} - P\{N(t) \geq n+1\} \\ &= P\{SS_n \leq t\} - P\{SS_{n+1} \leq t\} \\ &= G_n(t) - G_{n+1}(t), \end{aligned}$$

where $G_n(t)$ is the convolution of the inter-arrival times $F_1, F_2, F_3, \dots, F_n$. In other words,

$$G_n(t) = P\{F_1 + F_2 + \dots + F_n \leq t\}.$$

If the mean value of $N(t)$ is defined as the renewal function $m(t)$, then

$$\begin{aligned} m(t) &= E[N(t)] \\ &= \sum_{n=1}^{\infty} P\{N(t) \geq n\} \\ &= \sum_{n=1}^{\infty} P\{SS_n \leq t\} \\ &= \sum_{n=1}^{\infty} G_n(t). \end{aligned} \quad (8.35)$$

The derivative of $m(t)$ is known as the renewal density

$$\lambda(t) = m'(t).$$

In renewal theory, random variables representing inter-arrival distributions assume only non-negative values, and the Laplace transform of its distribution $F_1(t)$ is defined by

$$\mathcal{L}\{F_1(s)\} = \int_0^{\infty} e^{-sx} dF_1(x).$$

Therefore,

$$\mathcal{L}F_n(s) = \int_0^{\infty} e^{-\alpha^{n-1}st} dF_1(t) = \mathcal{L}F_1(\alpha^{n-1}s)$$

and

$$\begin{aligned} \mathcal{L}m_n(s) &= \sum_{n=1}^{\infty} \mathcal{L}G_n(s) \\ &= \sum_{n=1}^{\infty} \mathcal{L}F_1(s) \mathcal{L}F_1(\alpha s) \dots \mathcal{L}F_1(\alpha^{n-1}s). \end{aligned} \quad (8.36)$$

Since there is a one-to-one correspondence between distribution functions and its Laplace transform, it follows that the first inter-arrival distribution of a quasi-renewal process uniquely determines its renewal function.

If the inter-arrival time represents the error-free time (time to first failure), a quasi-renewal process can be used to model reliability growth in both software and hardware.

Suppose that all software faults have the same chance of being detected. If the inter-arrival time of a quasi-renewal process represents the error-free time of a software system, then the expected number of software faults in the time interval $[0, t]$ can be defined by the renewal function, $m(t)$, with parameter $\alpha > 1$. Denoted by $m_r(t)$, the number of remaining software faults at time t , it follows that

$$m_r(t) = m(T_c) - m(t),$$

where $m(T_c)$ is the number of faults that will eventually be detected through a software lifecycle T_c .

8.3.4 Nonhomogeneous Poisson Processes

The nonhomogeneous Poisson process model (NHPP), which represents the number of failures experienced up to time t , is a nonhomogeneous Poisson process $\{N(t) \text{ with } t \geq 0\}$. The main issue with the NHPP model is to determine an appropriate mean value function to denote the expected number of failures experienced up to a certain time.

Different assumptions mean that the model will end up with different functional forms of the mean value function. Note that the exponential assumption for the inter-arrival time between failures is relaxed in a renewal process, and the stationary assumption is relaxed in the NHPP.

The NHPP model is based on the following assumptions:

- The failure process has an independent increment; in other words, the number of failures during the time interval $(t, t+s)$ depends on the current time t and the length of the time interval s , and does not depend on the past history of the process.
- The failure rate of the process is given by

$$\begin{aligned} &P\{\text{exactly one failure in } (t, t+\Delta t)\} \\ &= P\{N(t+\Delta t) - N(t) = 1\} \\ &= \lambda(t)\Delta t + o(\Delta t), \end{aligned}$$

where $\lambda(t)$ is the intensity function.

- During a small interval Δt , the probability of more than one failure is negligible; that is,

$$P\{\text{two or more failures in } (t, t + \Delta t)\} = o(\Delta t),$$

- The initial condition is $N(0) = 0$.

Based on these assumptions, the probability that exactly n failures occur during the time interval $(0, t)$ for the NHPP is given by

$$\Pr\{N(t) = n\} = \frac{[m(t)]^n}{n!} e^{-m(t)} \quad n = 0, 1, 2, \dots, \quad (8.37)$$

where $m(t) = E[N(t)] = \int_0^t \lambda(s) ds$ and $\lambda(t)$ is the intensity function. It is easily shown that the mean value function $m(t)$ is nondecreasing.

The reliability $R(t)$, defined as the probability that there are no failures in the time interval $(0, t)$, is given by

$$R(t) = P\{N(t) = 0\} = e^{-m(t)}. \quad (8.38)$$

In general, the reliability $R(x | t)$ – the probability that there are no failures in the interval $(t, t + x)$ – is given by

$$R(x | t) = P\{N(t+x) - N(t) = 0\} = e^{-[m(t+x) - m(t)]}$$

and its density is given by

$$f(x) = \lambda(t+x) e^{-[m(t+x) - m(t)]},$$

where

$$\lambda(x) = \frac{\partial}{\partial x} [m(x)].$$

The variance of the NHPP can be obtained as follows:

$$\text{Var}[N(t)] = \int_0^t \lambda(s) ds$$

and the autocorrelation function is given by

$$\begin{aligned} \text{Cor}[s] &= E[N(t)] E[N(t+s) - N(t)] + E[N^2(t)] \\ &= \int_0^t \lambda(s) ds \int_0^{t+s} \lambda(s) ds + \int_0^t \lambda(s) ds \\ &= \int_0^t \lambda(s) ds \left[1 + \int_0^{t+s} \lambda(s) ds \right]. \end{aligned} \quad (8.39)$$

Example 8.5 Assume that the intensity λ is a random variable with pdf $f(\lambda)$. Then the probability that exactly n failures occur during the time interval $(0, t)$ is given by

$$P\{N(t) = n\} = \int_0^\infty e^{-\lambda t} \frac{(\lambda t)^n}{n!} f(\lambda) d\lambda. \quad (8.40)$$

If the pdf $f(\lambda)$ is given as the following gamma density function with parameters k and m :

$$f(\lambda) = \frac{1}{\Gamma(m)} k^m \lambda^{m-1} e^{-k\lambda} \quad \text{for } \lambda \geq 0 \quad (8.41)$$

then it can be shown that

$$P\{N(t) = n\} = \binom{n+m-1}{n} p^m q^n \quad n = 0, 1, 2, \dots \quad (8.42)$$

(this is also called a negative binomial density function), where

$$p = \frac{k}{t+k} \quad \text{and} \quad q = \frac{t}{t+k} = 1 - p. \quad (8.43)$$

8.4 Further Reading

The reader interested in a deeper understanding of advanced probability theory and stochastic processes should note the following citations, which refer to highly recommended books: Feller [3]; Pinsky and Karlin [4], Parzen [5], Melsa and Sage [6].

Appendix A: Laplace Transformation Functions

Let X be a nonnegative life time having probability density function f . The Laplace transform of a function $f(x)$, denote f^* , is defined as

$$\ell\{f(x)\} = f^*(s) = \int_0^\infty e^{-sx} f(x) dx \quad \text{for } s \geq 0. \quad (8.44)$$

The function f^* is called the Laplace transform of the function f . The symbol ℓ in Eq. (8.44) is called the Laplace transform operator. Note that $f^*(0) = 1$. By taking a differential derivative of $f^*(s)$, we obtain

$$\frac{\partial f^*(s)}{\partial s} = - \int_0^\infty x e^{-sx} f(x) dx.$$

Substitute $s = 0$ into the above equation, the first derivative of f^* , it yields the negative of the expected value of X or the first moment of X :

$$\left. \frac{\partial f^*(s)}{\partial s} \right|_{s=0} = -E(X).$$

Similarly, the second derivative yields the second moment of X when $s = 0$, that is,

$$\left. \frac{\partial f^*(s)}{\partial s} \right|_{s=0} = \int_0^\infty x^2 e^{-sx} f(x) dx \Big|_{s=0} = E(X^2).$$

In general, it can be shown that

$$\left. \frac{\partial^n f^*(s)}{\partial^n s} \right|_{s=0} = \int_0^\infty (-x)^n e^{-sx} f(x) dx \Big|_{s=0} = (-1)^n E(X^n)$$

Note that

$$e^{-sx} = \sum_{n=0}^\infty \frac{(-sx)^n}{n!}$$

then $f^*(s)$ can be rewritten as

$$f^*(s) = \sum_{n=0}^\infty \frac{(-s)^n}{n!} \mu_n,$$

where

$$\mu_n = (-1)^n \left. \frac{\partial^n f^*(s)}{\partial^n s} \right|_{s=0} = \int_0^\infty x^n e^{-sx} f(x) dx \Big|_{s=0} = E(X^n).$$

We can easily show that ℓ is a linear operator, that is

$$\ell \{c_1 f_1(x) + c_2 f_2(x)\} = c_1 \ell \{f_1(x)\} + c_2 \ell \{f_2(x)\}.$$

If $\ell\{f(t)\} = f^*(s)$, then we call $f(t)$ the inverse Laplace transform of $f^*(s)$ and write $\ell^{-1}\{f^*(s)\} = f(t)$. A summary of some common Laplace transform functions is listed in Table 8.1.

Example 8.6 Use the Laplace transforms to solve the following

$$\frac{\partial f(t)}{\partial t} + 3f(t) = e^{-t}, \tag{8.45}$$

with an initial condition: $f(0) = 0$. Obtain the solution $f(t)$. Here the Laplace transforms of $\frac{\partial f(t)}{\partial t}$, $f(t)$, and e^{-t} are

$$sf^*(s) - f(0), f^*(s), \text{ and } \frac{1}{s+1}$$

respectively. Thus, the Laplace transform of Eq. (8.45) is given by

$$sf^*(s) - f(0) + 3f^*(s) = \frac{1}{s+1}.$$

Since $f(0) = 0$ we have

$$(s+3)f^*(s) = \frac{1}{s+1} \text{ or } f^*(s) = \frac{1}{(s+1)(s+3)}.$$

Table 8.1 List of common Laplace transforms

$f(t)$	$\ell\{f(t)\} = f^*(s)$
$f^*(s)$	$f^*(s) = \int_0^\infty e^{-st} f(t) dt$
$\frac{\partial f(t)}{\partial t}$	$sf^*(s) - f(0)$
$\frac{\partial^2}{\partial t^2} [f(t)]$	$s^2 f^*(s) - sf(0) - \frac{\partial}{\partial t} f(0)$
$\frac{\partial^n}{\partial t^n} [f(t)]$	$s^n f^*(s) - s^{n-1} f(0) - \dots - \frac{\partial^{n-1}}{\partial t^{n-1}} f(0)$
$f(at)$	$\frac{1}{a} f^*\left(\frac{s}{a}\right)$
1	$\frac{1}{s}$
t	$\frac{1}{s^2}$
a	$\frac{a}{s}$
e^{-at}	$\frac{1}{s+a}$
te^{at}	$\frac{1}{(s-a)^2}$
$(1+at)e^{at}$	$\frac{s}{(s-a)^2}$
$\frac{1}{a} e^{-\frac{t}{a}}$	$\frac{1}{(1+sa)}$
t^p for $p > -1$	$\frac{\Gamma(p+1)}{s^{p+1}}$ for $s > 0$
t^n $n = 1, 2, 3, \dots$	$\frac{n!}{s^{n+1}}$ $s > 0$
$\frac{1}{a} (1 - e^{-at})$	$\frac{1}{s(s+a)}$
$\frac{1}{a} (e^{at} - 1)$	$\frac{1}{s(s-a)}$
$\frac{1}{a^2} (e^{-at} + at - 1)$	$\frac{1}{s^2(s+a)}$
$\frac{1}{b-a} (e^{-at} - e^{-bt})$	$\frac{1}{(s+a)(s+b)}$ $a \neq b$
$\frac{(ae^{at} - be^{bt})}{a-b}$	$\frac{s}{(s-a)(s-b)}$ $a \neq b$
$\frac{\alpha^k t^{k-1} e^{-\alpha t}}{\Gamma(k)}$	$\left(\frac{\alpha}{\alpha+s}\right)^k$

From Table 8.1, the inverse transform is

$$f(t) = \frac{1}{3-1} (e^{-t} - e^{-3t}) = \frac{1}{2} (e^{-t} - e^{-3t}). \tag{8.46}$$

Example 8.7 Let X be an exponential random variable with constant failure rate λ , that is, $f(x) = \lambda e^{-\lambda x}$ then we have

$$f^*(s) = \int_0^\infty \lambda e^{-sx} e^{-\lambda x} dx = \frac{\lambda}{s+\lambda}. \tag{8.47}$$

If X and Y are two independent random variables that represent life times with densities f_1 and f_2 , respectively, then the total life time's Z of those two X and Y , says $Z = X + Y$, has a pdf g that can be obtained as follows

$$g(z) = \int_0^z f_1(x) f_2(z-x) dx.$$

The Laplace transform of g in terms of f_1 and f_2 can be written as

$$\begin{aligned}
 g^*(s) &= \int_0^\infty e^{-sz} g(z) dz = \int_0^\infty \int_0^z e^{-sz} f_1(x) f_2(z-x) dx dz \\
 &= \int_0^\infty e^{-sx} f_1(x) dx \int_x^\infty e^{-s(z-x)} f_2(z-x) dz \\
 &= f_1^*(s) f_2^*(s).
 \end{aligned}
 \tag{8.48}$$

Example 8.8 If X and Y are both independent having the following pdfs: $f_1(x) = \lambda e^{-\lambda x}$ and $f_2(y) = \lambda e^{-\lambda y}$ for $x, y \geq 0$ and $\lambda \geq 0$ then we have

$$g^*(s) = f_1^*(s) f_2^*(s) = \left(\frac{\lambda}{s + \lambda} \right)^2.
 \tag{8.49}$$

From the Laplace transform Table 8.1, the inverse transform to solve for $g(z)$ is

$$g(z) = \frac{\lambda^2 t e^{-\lambda t}}{\Gamma(2)}$$

which is a special case of gamma pdf.

From Eq. (8.48), one can easily show the Laplace transform of the density function g_n of the total life time S_n of n independent life time's X_i with their pdf f_i for $i = 1, 2, \dots, n$, that

$$g_n^*(s) = f_1^*(s) f_2^*(s) \dots f_n^*(s) = \prod_{i=1}^n f_i^*(s)
 \tag{8.50}$$

If the pdf of n life time X_1, X_2, \dots, X_n are independent and identically distributed (i.i.d.) having a constant failure rate λ , then

$$g_n^*(s) = (f^*(s))^n = \left(\frac{\lambda}{s + \lambda} \right)^n.$$

From the Laplace transform table, we obtain the inverse transform for the solution function g as follows

$$g_n(z) = \frac{\lambda^n t^{n-1} e^{-\lambda t}}{\Gamma(n)}.
 \tag{8.51}$$

References

1. Pham, H.: Software Reliability. Springer, Berlin, Heidelberg (2000)
2. Wang, H., Pham, H.: A quasi renewal process and its applications in imperfect maintenance. Int. J. Syst. Sci. **27**(10), 1055–1062 (1996)
3. Feller, W.: An Introduction to Probability Theory and Its Applications, 3rd edn. Wiley, New York (1994)
4. Pinsky, M., Karlin, S.: Introduction to Stochastic Modeling, 4th edn. Academic Press (2010)
5. Parzen, E.: Stochastic Processes. SIAM (1987)
6. Melsa, J.L., Sage, A.P.: An Introduction to Probability and Stochastic Processes. Dover, Mineola; New York (2013)



Hoang Pham is a Distinguished Professor and former Chairman of the Department of Industrial & Systems Engineering at Rutgers University. He is the author or coauthor of 7 books and has published over 200 journal articles, 100 conference papers, and edited 20 books. His numerous awards include the 2009 IEEE Reliability Society *Engineer of the Year Award*. He is a Fellow of the IEEE and IISE.



Progressive Censoring Methodology

9

Narayanaswamy Balakrishnan and Erhard Cramer

Contents

9.1 Introduction and Fundamental Models	153
9.1.1 Basic Ideas of Progressive Censoring.....	153
9.1.2 Notation.....	155
9.1.3 Organization of the Paper.....	155
9.2 Progressive Type II Censoring	156
9.2.1 Probabilistic Results.....	156
9.2.2 Inference.....	161
9.2.3 Experimental Design.....	164
9.2.4 Connection of Progressive Type II Censoring to Coherent Systems.....	164
9.2.5 Connection of Progressive Type II Censoring to Ordered Pooled Samples.....	164
9.3 Progressive Type I Censoring	165
9.3.1 Distributional Results for Cohen's Progressive Censoring with Fixed Censoring Times.....	165
9.3.2 Distributional Results for Progressive Type I Censoring.....	166
9.3.3 Inference.....	167
9.4 Sampling Models Based on Progressively Censored Data	167
9.4.1 Progressive (Type I) Interval Censoring.....	167
9.4.2 Progressive Hybrid Censoring.....	168
9.4.3 Adaptive Progressive Censoring.....	169
9.4.4 Reliability and Stress-Strength Reliability.....	169
9.4.5 Competing Risks.....	170
9.4.6 Applications to System Data.....	170
9.4.7 Applications in Quality Control.....	171
9.4.8 Accelerated Life Testing.....	171
9.4.9 Stage Life Testing.....	171
9.4.10 Joint Progressive Censoring Schemes.....	173
References	173

Abstract

Progressive censoring has received great attention in the last decades especially in life testing and reliability. This review highlights fundamental applications, related models, and probabilistic and inferential results for progressively censored data. Based on the fundamental models of progressive type I and type II censoring, we present related models like adaptive and hybrid censoring as well as, e.g., stress-strength and competing risk models for progressively censored data. Focusing on exponentially and Weibull distributed lifetimes, an extensive bibliography emphasizing recent developments is provided.

Keywords

Censoring models · Ordered data · Hybrid censoring · Exponential distribution · Weibull distribution · Exact statistical inference · Lifetime analysis · Reliability · Step-stress testing · Competing risks · Experimental design · Ageing notions

9.1 Introduction and Fundamental Models

Monograph-length accounts on progressive censoring methodology have been provided by Balakrishnan and Cramer [1] and Balakrishnan and Aggarwala [2], while detailed reviews are due to [3] and [4]. In particular, [1] provides an up-to-date account to progressive censoring including many references and detailed explanations. Therefore, we provide essentially the basic models and results in the following, accompanied by recent developments and references which are not covered in the mentioned monograph.

N. Balakrishnan (✉)
Department of Mathematics and Statistics, McMaster University,
Hamilton, ON, Canada
e-mail: bala@mcmaster.ca

E. Cramer
Institute of Statistics, RWTH Aachen University, Aachen, Germany
e-mail: Erhard.Cramer@rwth-aachen.de

9.1.1 Basic Ideas of Progressive Censoring

According to [5], a progressively censored life testing experiment is conducted as follows. n items are put simultaneously on a test. At times $\tau_1 < \dots < \tau_m$, some items are randomly chosen among the surviving ones and removed from the experiment (see Fig. 9.1). In particular, at time τ_j , R_j items are withdrawn from the experiment. Originally, [5] had introduced two versions of progressive censoring, called type I and type II progressive censoring. In progressive type I censoring, the censoring times $\tau_1 < \dots < \tau_m$ are assumed to be fixed in advance (e.g., as prefixed inspection or maintenance times). For a better distinction, fixed censoring times are subsequently denoted by $T_1 < \dots < T_m$. Moreover, the censoring plan $\mathcal{R} = (R_1, \dots, R_m)$ is prespecified at the start of the life test. But, as failures occur randomly, it may happen that at some censoring time T_j , less than R_j items have survived. In that case, all the remaining items are withdrawn, and the life test is terminated at T_j . Notice that, due to this construction, observations beyond the largest censoring time T_m are possible. At this point, it is worth mentioning that the understanding of progressive type I censoring has changed over time. As has been noted in [6], the understanding of the term *progressive type I censoring* has been used differently after the publication of monograph [2] [see also 7]. Since then, right censoring has been considered as a feature of progressive type I censoring, that is, T_m is considered as a

termination time of the experiment (see Fig. 9.2). Therefore, in progressive type I censoring, we distinguish the initially planned censoring plan $\mathcal{R}^0 = (R_1^0, \dots, R_{m-1}^0)$ from the effectively applied one denoted by $\mathcal{R} = (R_1, \dots, R_{m-1})$. Notice that we drop the m th censoring number R_m in the plan since it is always random due to the right censoring. In order to distinguish these scenarios, [6] called the original scenario *Cohen's progressive censoring scheme with fixed censoring times*.

The second version of progressive censoring proposed by Cohen [5] is called progressive type II censoring which may be considered as the most popular version of progressive censoring. Here, the censoring times are induced by the lifetimes of the surviving units in the sense that the next withdrawal is carried out the first failure after the removal of items. Suppose the items are numbered by $1, \dots, n$ with lifetimes X_1, \dots, X_n and denote by the set \mathcal{R}_j the numbers of the items available before the j th removal. Then, the next removal time is defined by $X_{j:m:n} = \min_{i \in \mathcal{R}_j} X_i$ (see Fig. 9.3), $j = 1, \dots, m$. Clearly, $\mathcal{R}_1 = \{1, \dots, n\}$ and $X_{1:m:n} = X_{1:n}$ is given by the minimum of the lifetimes. Furthermore, $|\mathcal{R}_j| = n - j + 1 - \sum_{i=1}^{j-1} R_i = \gamma_j$, $j = 1, \dots, m$. The censoring times are iteratively constructed and random so that they are not known in advance (in contrast to the type I censoring scheme). However, the censoring plan and the sample size are fixed here. In fact, given n and m , the set of admissible progressive type II censoring plans is given by

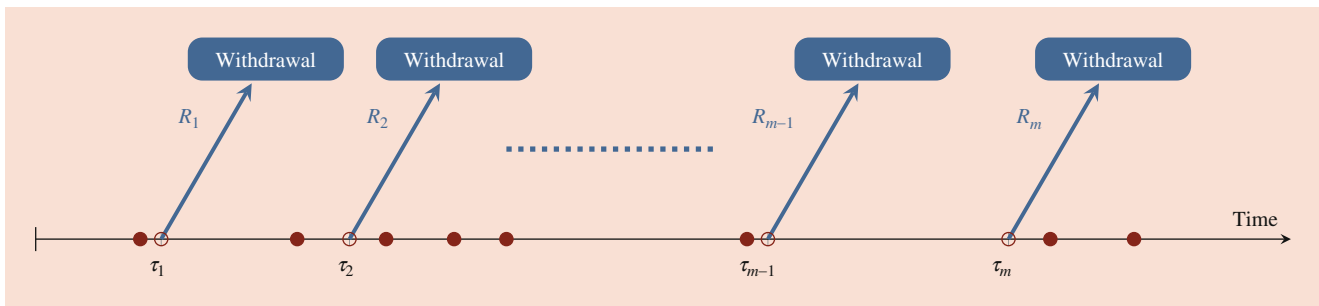


Fig. 9.1 Progressive censoring scheme with censoring times $\tau_1 < \dots < \tau_m$ and censoring plan $\mathcal{R} = (R_1, \dots, R_m)$

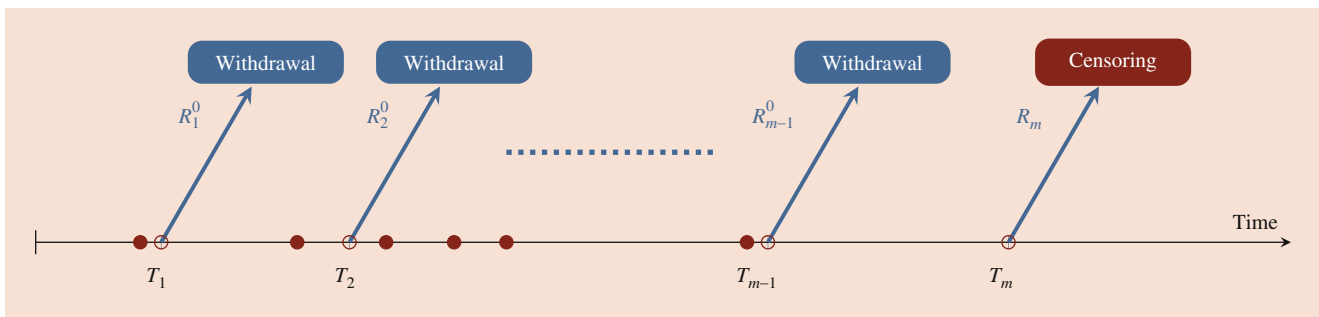


Fig. 9.2 Progressive type I censoring with censoring times $T_1 < \dots < T_{m-1}$, time censoring at T_m , and initial censoring plan $\mathcal{R}^0 = (R_1^0, \dots, R_{m-1}^0)$

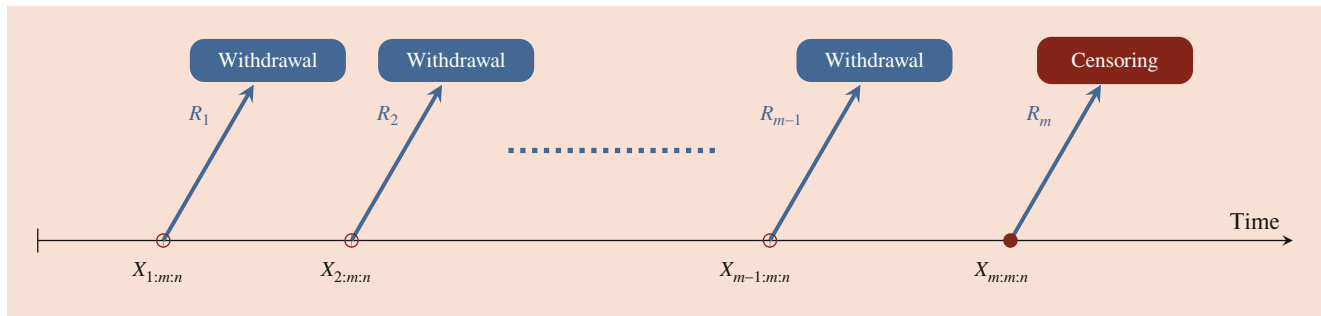


Fig. 9.3 Progressive type II censoring with censoring times $X_{1:m:n} < \dots < X_{m:m:n}$ and censoring plan $\mathcal{R} = (R_1, \dots, R_m)$

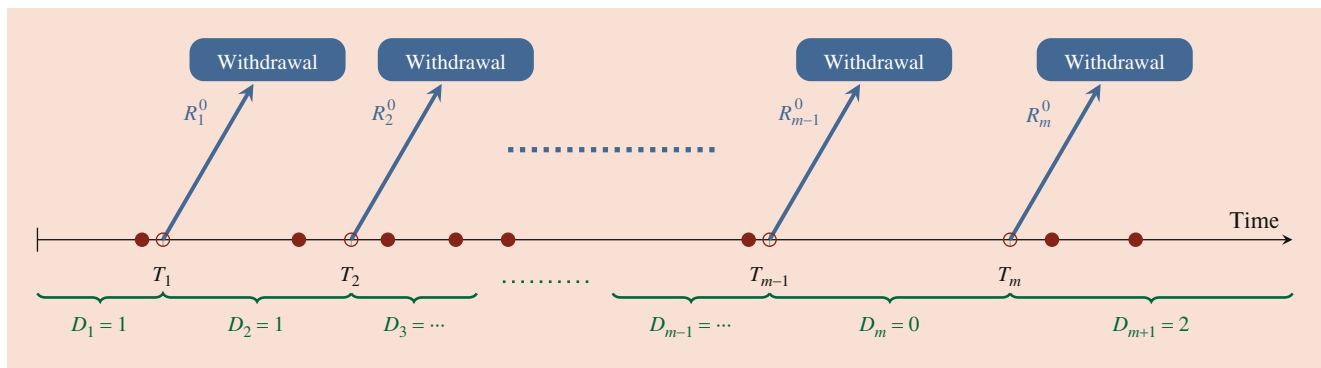


Fig. 9.4 Progressive interval censoring with censoring times $T_1 < \dots < T_m$, initial censoring plan $\mathcal{R}^0 = (R_1^0, \dots, R_m^0)$, and random counters D_1, \dots, D_{m+1}

$$\mathcal{C}_{m,n}^m = \left\{ (r_1, \dots, r_m) \in \mathbb{N}_0^m : \sum_{i=1}^m r_i = n - m \right\}. \quad (9.1)$$

Based on the above fundamental models, further versions of progressive censoring have been proposed. Progressive type I interval censoring uses only partial information from a progressively type I censored life test. In particular, it is assumed that only the number D_j of items failing in an interval $(T_{j-1}, T_j]$ is known (see Fig. 9.4 for Cohen’s progressive censoring with fixed censoring times). The corresponding situation under type I right censoring is depicted in Fig. 9.5.

The above censoring schemes have been extended in various directions. For instance, a wide class of models called progressive hybrid censoring has been generated by combining type I and type II censoring procedures. In type I progressive hybrid censoring, a type I censoring mechanism has been applied to progressively type II censored data $X_{1:m:n}, \dots, X_{m:m:n}$ by Childs et al. [8] [see also 9, 10], extending a model of [11] by introducing a threshold T . The resulting time censored data $X_{j:m:n}^{h,I} = \min\{X_{j:m:n}, T\}$, $j = 1, \dots, m$, will be discussed further in Sect. 9.4.2. Considering the so-called extended progressively type II censored sample by dropping the right censoring (see (9.15)), that is,

$X_{1:m+R_m:n}, \dots, X_{m+R_m:m+R_m:n}$, a type II progressively hybrid censored sample can be defined by the condition $X_{K:m+R_m:n} \leq T < X_{K+1:m+R_m:n}$, $m \leq K \leq m + R_m$. Further versions have been summarized in [12]. An extensive survey on (progressive) hybrid censoring schemes is provided in the recent monograph by [380].

Motivated by the Ng-Kundu-Chan model introduced in [13], adaptive progressive censoring schemes have been proposed by Cramer and Iliopoulos [14] and Cramer and Iliopoulos [15]. In these models, censoring plans and censoring times may be chosen adaptively according to the observed data. Such models are presented briefly in Sect. 9.4.3.

In most cases, progressive censoring is studied under the assumption that the underlying lifetimes X_1, \dots, X_n are independent and identically distributed (iid) random variables. If not noted explicitly, all results presented in the following are based on this assumption. However, relaxations of this assumption have been made. For instance, [16] discussed the case of heterogeneous distributions, that is, $X_i \sim F_i$, $1 \leq i \leq n$, are independent random variables but may have a different cumulative distribution function (see also [17, 18]). Rezapour et al. [19, 20] assumed dependent underlying lifetimes. For a review, we refer to [1, Chapter 10].

Notation	Explanation
cdf	Cumulative distribution function
pdf	(Probability) density function
iid	Independent and identically distributed
BLUE	Best linear unbiased estimator
MLE	Maximum likelihood estimator
UMVUE	Uniformly minimum variance unbiased estimator
PHR model	Proportional hazard rates model
\mathbb{R}	Set of real numbers
\mathbb{N}, \mathbb{N}_0	Set of positive and nonnegative integers, respectively
F^{\leftarrow}	Quantile function of a cdf F
$\mathbb{I}_A(t)$	Indicator function for a set A ; $\mathbb{I}_A(t) = 1$ for $t \in A$, $\mathbb{I}_A(t) = 0$, otherwise
$\stackrel{d}{=}$	Identical in distribution
\mathbf{x}_m	$\mathbf{x}_m = (x_1, \dots, x_m)$
$\mathbf{x}_n \wedge \mathbf{y}_n$	$\mathbf{x}_n \wedge \mathbf{y}_n = (\min(x_1, y_1), \dots, \min(x_n, y_n))$
$\mathbf{x}_n \vee \mathbf{y}_n$	$\mathbf{x}_n \vee \mathbf{y}_n = (\max(x_1, y_1), \dots, \max(x_n, y_n))$
$[x]_+$	$\max(x, 0)$
$d_{\bullet m}$	$d_{\bullet m} = \sum_{i=1}^m d_i$ for $d_1, \dots, d_m \in \mathbb{R}$
\mathcal{R}	Censoring plan $\mathcal{R} = (R_1, \dots, R_m)$ with censoring numbers R_1, \dots, R_m
$\mathcal{C}_{m,n}^m$	Set of admissible (progressive type II) censoring plans defined in (9.1)
$(\gamma_1, \dots, \gamma_m)$	$\gamma_i = \sum_{j=i}^m (R_j + 1)$, $1 \leq i \leq m$, for a censoring plan $\mathcal{R} = (R_1, \dots, R_m)$
$X_{1:m:n}, \dots, X_{m:m:n}, X_{1:m:n}^{\mathcal{R}}, \dots, X_{m:m:n}^{\mathcal{R}}$	Progressively type II censored order statistics based on a sample X_1, \dots, X_n and censoring plan \mathcal{R}
$\mathbf{X}^{\mathcal{R}}$	$\mathbf{X}^{\mathcal{R}} = (X_{1:m:n}^{\mathcal{R}}, \dots, X_{m:m:n}^{\mathcal{R}})$
$U_{1:m:n}, \dots, U_{m:m:n}$	Uniform progressively type II censored order statistics
$X_{1:K:n}^I, \dots, X_{K:K:n}^I$	Progressively type I censored order statistics or progressively censored order statistics with fixed censoring times based on a sample X_1, \dots, X_n
$\mathbf{X}^{I, \mathcal{R}}$	$\mathbf{X}^{I, \mathcal{R}} = (X_{1:K:n}^I, \dots, X_{K:K:n}^I)$
$X_{1:n}, \dots, X_{n:n}$	Order statistics based on a sample X_1, \dots, X_n
$\text{Exp}(\mu, \vartheta)$	Two-parameter exponential distribution with pdf $f(t) = \frac{1}{\vartheta} e^{-(t-\mu)/\vartheta}$, $t > \mu$
$\text{Exp}(\vartheta) = \text{Exp}(0, \vartheta)$	Exponential distribution with mean ϑ and pdf $f(t) = \frac{1}{\vartheta} e^{-t/\vartheta}$, $t > 0$
F_{exp}	cdf of standard exponential distribution $\text{Exp}(1)$; $F_{\text{exp}}(t) = 1 - e^{-t}$, $t \geq 0$
$\text{Wei}(\vartheta, \beta)$	Weibull distribution with parameters $\vartheta, \beta > 0$ and pdf $f(t) = \frac{\beta}{\vartheta} t^{\beta-1} e^{-t^\beta/\vartheta}$, $t > 0$
$U(0, 1)$	Uniform distribution on the interval $(0, 1)$
$\chi^2(r)$	χ^2 -distribution with r degrees of freedom
$X \sim F$	X is distributed according to a cdf F
$X_1, \dots, X_n \stackrel{\text{iid}}{\sim} F$	X_1, \dots, X_n are independent and identically distributed according to a cdf F

9.1.2 Notation

Throughout, we use the following notation and abbreviations.

9.1.3 Organization of the Paper

In the following sections, we discuss the most popular versions of progressive censoring in detail, that is, progressive type II censoring (Sect. 9.2) and progressive type I censoring (Sect. 9.3). Further, progressive censoring with fixed censoring times is also addressed in Sect. 9.3. In Sect. 9.4, related data like progressive interval censoring, progressive hybrid censoring, and adaptive progressive censoring as well as applications in reliability and lifetime analysis are discussed. Due to their importance, we focus on exponentially and

Weibull distributed lifetimes. Except when otherwise stated, the underlying lifetimes X_1, \dots, X_n are supposed to be independent and identically distributed according to a cdf F , that is, $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} F$.

9.2 Progressive Type II Censoring

9.2.1 Probabilistic Results

Fundamental tools in studying properties of progressively type II censored order statistics are the joint pdf of $X_{1:m:n}, \dots, X_{m:m:n}$ and the quantile representation. The joint pdf of progressively type II censored order statistics $X_{1:m:n}, \dots, X_{m:m:n}$ based on an (absolutely continuous) cdf F with pdf f is given by

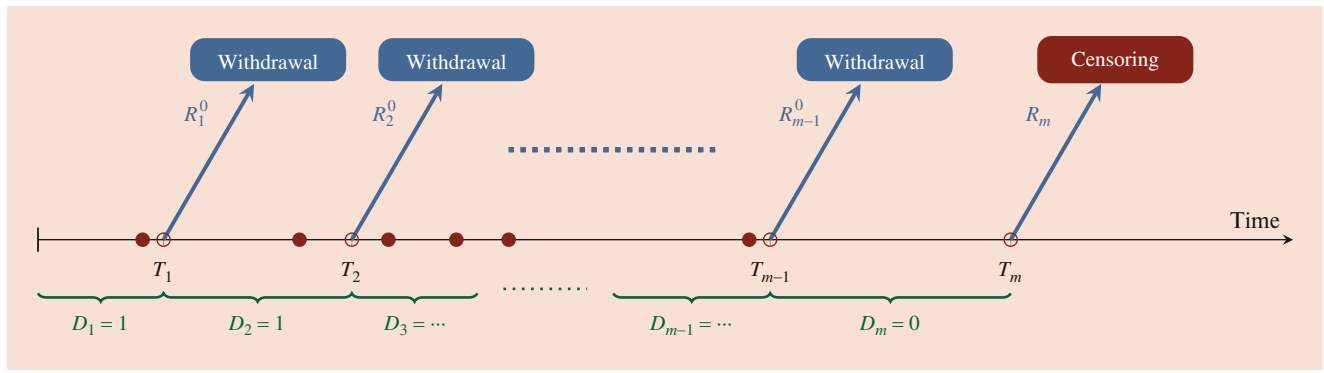


Fig. 9.5 Progressive type I interval censoring with censoring times $T_1 < \dots < T_{m-1}$, time censoring at T_m , initial censoring plan $\mathcal{R}^0 = (R_1^0, \dots, R_{m-1}^0)$, and random counters D_1, \dots, D_m

$$f^{X^{\mathcal{R}}}(\mathbf{x}_m) = \begin{cases} \prod_{j=1}^m [\gamma_j f(x_j)(1 - F(x_j))^{R_j}], & x_1 \leq \dots \leq x_m \\ 0, & \text{otherwise} \end{cases} \quad (9.2)$$

where $\gamma_j = \sum_{i=j}^m (R_i + 1)$ denotes the number of items remaining in the experiment before the j th failure, $1 \leq j \leq m$. Notice that $n = \gamma_1 > \dots > \gamma_m \geq 1$. It is immediate from (9.2) that progressively type II censored order statistics are connected to the distributional model of generalized order statistics (see [21–23]) which covers progressively type II censored order statistics as a particular case (for details, see [1], Section 2.2).

Exponential Distributions

From (9.2), the pdf for an exponential population $\text{Exp}(\mu, \vartheta)$ can be directly obtained, that is, the pdf of exponential progressively type II censored order statistics $\mathbf{X}^{\mathcal{R}} = (X_{1:m:n}, \dots, X_{m:m:n})$ is given by

$$f^{X^{\mathcal{R}}}(\mathbf{x}_m) = \frac{1}{\vartheta^m} \left(\prod_{j=1}^m \gamma_j \right) \exp\left(-\frac{1}{\vartheta} \sum_{j=1}^m (R_j + 1)(x_j - \mu)\right),$$

$$\mu \leq x_1 \leq \dots \leq x_m.$$

This expression is important in deriving, e.g., properties of exponential progressively type II censored order statistics as well as in developing statistical inference. As pointed out by Thomas and Wilson [24] (see also [25]), the normalized spacings $S_j^{\mathcal{R}} = \gamma_j(X_{j:m:n} - X_{j-1:m:n})$, $1 \leq j \leq m$, with $X_{0:m:n} = \mu$ defined as left endpoint of support are iid random variables, that is (see [1], Theorem 2.3.2),

$$S_1^{\mathcal{R}}, \dots, S_m^{\mathcal{R}} \stackrel{\text{iid}}{\sim} \text{Exp}(\vartheta). \quad (9.3)$$

On the other hand, exponential progressively type II censored order statistics can be written in terms of their spacings yielding the identity

$$X_{r:m:n} = \mu + \sum_{i=1}^r \frac{1}{\gamma_i} S_i^{\mathcal{R}} = X_{r-1:m:n} + \frac{1}{\gamma_r} S_r^{\mathcal{R}}, \quad 1 \leq r \leq m, \quad (9.4)$$

where $X_{0:m:n} = \mu$. This representation allows us to derive many properties of exponential progressively type II censored order statistics. For instance, using (9.3) and (9.4), the one-dimensional marginal pdfs and cdfs are given by (see also [26])

$$f^{X_{r:m:n}}(t) = \left(\prod_{j=1}^r \gamma_j \right) \sum_{j=1}^r a_{j,r} e^{-\gamma_j t},$$

$$F^{X_{r:m:n}}(t) = 1 - \left(\prod_{i=1}^r \gamma_i \right) \sum_{j=1}^r \frac{1}{\gamma_j} a_{j,r} e^{-\gamma_j t}, \quad t > 0,$$

where $a_{j,r} = \prod_{\substack{i=1 \\ i \neq j}}^r \frac{1}{\gamma_i - \gamma_j}$, $1 \leq j \leq r \leq n$. Representations of bivariate and arbitrary marginals can be found in [1, Section 2.4], (see also [27, 28]). Moreover, it follows from (9.4) that $X_{1:m:n}, \dots, X_{m:m:n}$ form a Markov chain, that is, for $2 \leq r \leq m$,

$$X_{r:m:n} \mid (X_{r-1:m:n} = x_{r-1}, \dots, X_{1:m:n} = x_1)$$

$$\stackrel{d}{=} X_{r:m:n} \mid (X_{r-1:m:n} = x_{r-1})$$

$$\stackrel{d}{=} \frac{1}{\gamma_r} S_r^{\mathcal{R}} + x_{r-1} \sim \text{Exp}(x_r, \vartheta / \gamma_r).$$

Furthermore, $X_{1:m:n} \stackrel{d}{=} \text{Exp}(\mu, \vartheta / \gamma_1)$. These representations allow direct calculation of moments. For instance, one gets

$$EX_{r:m:n} = \mu + \vartheta \sum_{j=1}^r \frac{1}{\gamma_j}, \quad \text{Var } X_{r:m:n} = \vartheta^2 \sum_{j=1}^r \frac{1}{\gamma_j^2},$$

$$\text{Cov}(X_{r:m:n}, X_{s:m:n}) = \text{Var } X_{r:m:n} = \vartheta^2 \sum_{j=1}^r \frac{1}{\gamma_j^2}, \quad 1 \leq r \leq s \leq m. \quad (9.5)$$

Further results on moments, e.g., higher order moments, existence of moments, bounds, and recurrence relations, can be found in [1] and the references cited therein.

Other Distributions

Probabilistic results for other distributions can be obtained from the pdf in (9.2) or, alternatively, from the results obtained for the exponential distribution and the following result due to [29,30] (for a proof, see [1]). It shows that most of the distributional results can be obtained for a uniform distribution and then transformed to an arbitrary cdf F .

Theorem 9.2.1 *Suppose $X_{1:m:n}, \dots, X_{m:m:n}$ and $U_{1:m:n}, \dots, U_{m:m:n}$ are progressively type II censored order statistics based on a cdf F and a uniform distribution, respectively. Then,*

$$(X_{j:m:n})_{1 \leq j \leq m} \stackrel{d}{=} (F^{\leftarrow}(U_{j:m:n}))_{1 \leq j \leq m}.$$

Theorem 9.2.1, together with the representation of the joint pdf for uniform distributions, yields an expression for progressively type II censored order statistics based on an arbitrary cdf F , that is,

$$F^{X_{r:m:n}}(t) = 1 - \left(\prod_{i=1}^r \gamma_i \right) \sum_{j=1}^r \frac{1}{\gamma_j} a_{j,r} (1 - F(t))^{\gamma_j}, \quad t \in \mathbb{R}.$$

If F is absolutely continuous with pdf f , then the pdf is given by

$$f^{X_{r:m:n}}(t) = \left(\prod_{j=1}^r \gamma_j \right) f(t) \sum_{j=1}^r a_{j,r} (1 - F(t))^{\gamma_j - 1}, \quad t \in \mathbb{R}.$$

Similar representations can be obtained for multiple progressively type II censored samples (see [1], p.41, [28,31]). Log-concavity and unimodality properties of the distributions are studied in [32–36].

From (9.4), it follows that $F_{\text{exp}}(X_{r:m:n}) = 1 - \prod_{j=1}^r (e^{-S_j^{\mathcal{R}}/\vartheta})^{1/\gamma_j}$ with $S_1^{\mathcal{R}}, \dots, S_m^{\mathcal{R}}$ as in (9.3). Thus, $U_j = e^{-S_j^{\mathcal{R}}/\vartheta}$, $1 \leq j \leq m$, are independent uniformly distributed random variables. This yields the identity

$$U_{r:m:n} \stackrel{d}{=} 1 - \prod_{j=1}^r U_j^{1/\gamma_j}, \quad 1 \leq r \leq m,$$

of $U_{r:m:n}$ as a product of independent random variables. Combining this expression with the quantile representation from Theorem 9.2.1, we arrive at the following representation [see also 37].

Theorem 9.2.2 *Let $X_{1:m:n}, \dots, X_{m:m:n}$ be progressively type II censored order statistics from an arbitrary cdf F and $U_1, \dots, U_m \stackrel{iid}{\sim} U(0, 1)$. Then,*

$$X_{1:m:n}, \dots, X_{m:m:n} \stackrel{d}{=} F^{\leftarrow} \left(1 - \prod_{j=1}^r U_j^{1/\gamma_j} \right), \quad 1 \leq r \leq m.$$

As has been noticed in [38], this representation provides an alternative method to simulate progressively type II censored order statistics from a cdf F (for a survey on simulation methods, see [1, Chapter 8]). Alternatively, one can also write

$$X_{r:m:n} \stackrel{d}{=} F^{\leftarrow} \left(F_{\text{exp}} \left(\sum_{j=1}^r \frac{1}{\gamma_j} Z_j \right) \right), \quad 1 \leq r \leq m,$$

with $Z_1, \dots, Z_m \stackrel{iid}{\sim} \text{Exp}(1)$. This result illustrates that $X_{1:m:n}, \dots, X_{m:m:n}$ form a Markov chain with transition probabilities

$$P(X_{r:m:n} \leq t \mid X_{r-1:m:n} = s) = 1 - \left(\frac{1 - F(t)}{1 - F(s)} \right)^{\gamma_r}, \quad s \leq t \text{ with } F(s) < 1.$$

Further results on the dependence structure of progressively type II censored order statistics are available. For instance, [28] has shown that progressively type II censored order statistics exhibit the MTP₂-property which implies that progressively type II censored order statistics are always positively correlated. The block independence property has been established by Iliopoulos and Balakrishnan [39]. In order to formulate the result, we introduce the number of progressively type II censored order statistics that do not exceed a threshold T , i.e., $D = \sum_{j=1}^m \mathbb{I}_{(-\infty, T]}(X_{j:m:n})$. Then, the probability mass function of D is given by the probabilities

$$\begin{aligned} P(D = 0) &= (1 - F(T))^n, \\ P(D = d) &= \left(\prod_{i=1}^d \gamma_i \right) \sum_{j=1}^{d+1} a_{j,d+1} (1 - F(T))^{\gamma_j}, \quad d = 1, \dots, m - 1, \\ P(D = m) &= F^{X_{m:m:n}}(T). \end{aligned}$$

Given $d \in \{1, \dots, m - 1\}$, a cdf F , and a censoring plan $\mathcal{R} = (R_1, \dots, R_m)$, the block independence property is as follows: Conditionally on $D = d$, the random vectors $(X_{1:m:n}, \dots, X_{d:m:n})$ and $(X_{d+1:m:n}, \dots, X_{m:m:n})$ are independent with

$$\begin{aligned} (X_{1:m:n}, \dots, X_{d:m:n}) &\stackrel{d}{=} (V_{1:d:\kappa_d}^{K_d}, \dots, V_{d:d:\kappa_d}^{K_d}), \\ (X_{d+1:m:n}, \dots, X_{m:m:n}) &\stackrel{d}{=} (W_{1:m-d:\gamma_d}, \dots, W_{m-d:m-d:\gamma_d}) \end{aligned} \tag{9.6}$$

where $\mathbf{K}_d = (K_1, \dots, K_d)$ is a random censoring plan on the Cartesian product $\prod_{j=1}^d \{0, \dots, R_j\}$ with probability mass function

$$p^{\mathbf{K}_d}(\mathbf{k}_d) = \frac{1}{P(D=d)} F^{\eta_1(d)}(T) (1-F(T))^{n-\eta_1(d)} \prod_{i=1}^d \frac{\gamma_i}{\eta_i(d)} \binom{R_i}{k_i},$$

$\kappa_d = \sum_{j=1}^d (1 + K_j)$, and $\eta_i(d) = \sum_{j=i}^d (k_j + 1)$, $1 \leq i \leq d$.
Further:

1. V_1, \dots, V_n are iid random variables with right truncated cdf F_T given by $F_T(t) = \frac{F(t)}{F(T)}$, $t \leq T$.
2. W_1, \dots, W_{γ_d} are iid random variables with left truncated cdf G_T given by $G_T(t) = 1 - \frac{1-F(t)}{1-F(T)}$, $t \geq T$.

The sample size κ_d of the progressively type II censored order statistics $V_{1:d:\kappa_d}^{\mathbf{K}_d}, \dots, V_{d:d:\kappa_d}^{\mathbf{K}_d}$ is a random variable. The above representation means that the distribution of $(X_{1:m:n}, \dots, X_{d:m:n})$, given $D = d$, is a mixture of distributions of progressively type II censored order statistics with mixing distribution $p^{\mathbf{K}_d}$. It is well known that right truncation of progressively type II censored order statistics does not result in progressively type II censored order statistics from the corresponding right truncated distribution [see, e.g., 2]. This is due to the fact that those observations (progressively) censored before T could have values larger than T .

Connection to Order Statistics and Other Models of Ordered Data

Order statistics [see, e.g., 40, 41] can be interpreted as special progressively type II censored order statistics by choosing the censoring plan $\mathcal{O} = (0, \dots, 0)$. Then, we have $m = n$ and $X_{j:n}^{\mathcal{O}} = X_{j:n}$, $1 \leq j \leq n$. Furthermore, the censoring plan $\mathcal{O}^* = (0, \dots, 0, R_m)$ with $n = m + R_m$ yields $X_{j:m:n}^{\mathcal{O}^*} = X_{j:m+R_m}$, $1 \leq j \leq m$, leading to a type II right censored sample. Thus, all the results developed for progressively type II censored order statistics can be specialized to order statistics. Detailed accounts to order statistics are provided by Arnold et al. [40] and David and Nagaraja [41].

As mentioned above, progressively type II censored order statistics can be seen as particular generalized order statistics and sequential order statistics, respectively (see [21, 23, 37]). In this regard, results obtained in these models hold also for progressively type II censored order statistics by specifying model parameters and distributions suitably. For pertinent details, we refer to the references given above.

Moments

Many results on moments have been obtained for progressively type II censored order statistics (see, e.g., [1], Chapter 7). This discussion includes explicit results for selected distributions, e.g., exponential distributions (see (9.5)), Weibull, Pareto, Lomax, reflected power, and extreme value distributions. Further topics are existence of moments (see [42, 43]), bounds (see, e.g., [32, 44, 45]), recurrence relations (see, e.g., [27, 46]), and approximations (see, e.g., [47]). Furthermore, the accurate computation of moments

has been discussed in [48] (see also [49]). It should be noted that an enormous number of papers have discussed moments as well as related recurrence relations for particular distributions.

Stochastic Orders and Stochastic Comparisons

Results on various stochastic orders of progressively type II censored order statistics have been mostly established in terms of generalized and sequential order statistics (see [1], Section 3.2). Therefore, the results can be applied to progressively type II censored order statistics by choosing particular parameter values. For information, we present definitions of the most important stochastic orders discussed for progressively type II censored order statistics. For a general discussion, we refer to [50, 51], and [52]. A review of results on multivariate stochastic orderings for generalized order statistics is provided in [53] (see also [1]).

Let $X \sim F$, $Y \sim G$ be random variables and let f and g denote the respective pdfs. For simplicity, it is assumed that the supports are subsets of the set of positive values. In the following, we present some selected results on stochastic orderings under progressive type II censoring and provide references for further reading.

Stochastic Order/Multivariate Stochastic Order

- (i) X is said to be stochastically smaller than Y , that is, $X \leq_{st} Y$ or $F \leq_{st} G$, iff $\bar{F}(x) \leq \bar{G}(x)$ for all $x \geq 0$.
- (ii) Let $\mathbf{X} = (X_1, \dots, X_n)'$, $\mathbf{Y} = (Y_1, \dots, Y_n)'$ be random vectors. Then, \mathbf{X} is said to be stochastically smaller than \mathbf{Y} , that is, $\mathbf{X} \leq_{st} \mathbf{Y}$ or $F^{\mathbf{X}} \leq_{st} F^{\mathbf{Y}}$, iff $E\phi(\mathbf{X}) \leq E\phi(\mathbf{Y})$ for all nondecreasing functions $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ provided the expectations exist.

Notice that, for $n = 1$, the definition of the multivariate stochastic order is equivalent to the (common) definition in the univariate case.

Belzunce et al. [54] has established the preservation of the stochastic order when the baseline distributions are stochastically ordered.

Theorem 9.2.3 *Let $\mathbf{X}^{\mathcal{R}}$ and $\mathbf{Y}^{\mathcal{R}}$ be vectors of progressively type II censored order statistics from continuous cdfs F and G with censoring plan \mathcal{R} , respectively. Then, for $F \leq_{st} G$, $\mathbf{X}^{\mathcal{R}} \leq_{st} \mathbf{Y}^{\mathcal{R}}$.*

A comparison in terms of the univariate stochastic order has been established by Khaledi [55] using the following partial ordering of γ -vectors (see [56, 57]). Let \mathcal{R} and \mathcal{S} be censoring plans with corresponding γ -values $\gamma_i(\mathcal{R}) = \sum_{k=i}^{m_1} (R_k + 1)$ and $\gamma_i(\mathcal{S}) = \sum_{k=i}^{m_2} (S_k + 1)$. For $1 \leq j \leq i$,

$$(\gamma_1(\mathcal{S}), \dots, \gamma_j(\mathcal{S})) \leq_p (\gamma_{i-j+1}(\mathcal{R}), \dots, \gamma_i(\mathcal{R})) \quad (9.7)$$

iff

$$\prod_{k=1}^{\ell} \gamma_{i-k+1}(\mathcal{R}) \leq \prod_{k=1}^{\ell} \gamma_k(\mathcal{S}) \text{ for } \ell = 1, \dots, j.$$

Theorem 9.2.4 Let F, G be continuous cdfs with $F \leq_{st} G$ and $X \sim F, Y \sim G$. Moreover, let $\mathcal{R} \in \mathcal{C}_{m_1, n_1}^{m_1}, \mathcal{S} \in \mathcal{C}_{m_2, n_2}^{m_2}$ with $m_1, m_2 \in \mathbb{N}$ be censoring plans. Then:

- (i) $X_{i:m_1:m_1}^{\mathcal{R}} \leq_{st} Y_{i:m_1:m_1}^{\mathcal{R}}, 1 \leq i \leq m_1$.
- (ii) If $1 \leq j \leq i$ and condition (9.7) holds, then $X_{j:m_2:m_2}^{\mathcal{S}} \leq_{st} Y_{i:m_1:m_1}^{\mathcal{R}}$.

Applications to stochastic ordering of spacings of progressively type II censored order statistics can be found in [58].

Failure Rate/Hazard Rate Order, Reversed Hazard Rate Order

- (i) X is said to be smaller than Y in the hazard rate order, that is, $X \leq_{hr} Y$ or $F \leq_{hr} G$, iff $\bar{F}(x)\bar{G}(y) \leq \bar{F}(y)\bar{G}(x)$ for all $0 \leq y \leq x$.
- (ii) X is said to be smaller than Y in the reversed hazard rate order, that is, $X \leq_{rh} Y$ or $F \leq_{rh} G$, iff $F(x)G(y) \leq F(y)G(x)$ for all $0 \leq y \leq x$.

For the hazard rate order, the ratio $\frac{\bar{F}(x)}{\bar{G}(x)}$ is nonincreasing in $x \geq 0$ where $\frac{a}{0}$ is defined to be ∞ . If F and G are absolutely continuous cdfs with pdfs f and g , respectively, then hazard rate ordering is equivalent to increasing hazard rates, that is,

$$\lambda_F(x) = \frac{f(x)}{1 - F(x)} \leq \frac{g(x)}{1 - G(x)} = \lambda_G(x) \text{ for all } x \geq 0.$$

For the reversed hazard rate order, the ratio $\frac{F(x)}{G(x)}$ is nonincreasing in $x \geq 0$, where $\frac{a}{0}$ is defined to be ∞ .

Results for (multivariate) hazard rate orders of progressively type II censored order statistics have been obtained by Belzunce et al. [54], Khaledi [55], and Hu and Zhuang [59]. For instance, replacing the stochastic order by the hazard rate order in Theorem 9.2.4, an analogous result is true (see [1, Theorem 3.2.3]).

Likelihood Ratio Order/Multivariate Likelihood Ratio Order

- (i) X is said to be smaller than Y in the likelihood ratio order, that is, $X \leq_{lr} Y$ or $F \leq_{lr} G$, iff $f(x)g(y) \leq f(y)g(x)$ for all $0 \leq y \leq x$.
- (ii) Let $\mathbf{X} = (X_1, \dots, X_n)'$, $\mathbf{Y} = (Y_1, \dots, Y_n)'$ be random vectors with pdfs f^X and f^Y . Then, \mathbf{X} is said to be smaller than \mathbf{Y} in the multivariate likelihood ratio order, that is, $\mathbf{X} \leq_{lr} \mathbf{Y}$ or $F^X \leq_{lr} F^Y$, iff

$$f^X(\mathbf{x}_n)f^Y(\mathbf{y}_n) \leq f^X(\mathbf{x}_n \wedge \mathbf{y}_n)f^Y(\mathbf{x}_n \vee \mathbf{y}_n)$$

for all $\mathbf{x} = (x_1, \dots, x_n)', \mathbf{y} = (y_1, \dots, y_n)' \in \mathbb{R}^n$.

The (multivariate) likelihood ratio order has been discussed, e.g., by Korwar [60], Hu and Zhuang [59], Cramer et al. [61], Belzunce et al. [54], Zhuang and Hu [62], Balakrishnan et al. [63], Sharafi et al. [64], and Arriaza et al. [65]. The following result is due to [60] (see [59] for generalized order statistics).

Theorem 9.2.5 Let $\mathcal{R} \in \mathcal{C}_{m_1, n_1}^{m_1}, \mathcal{S} \in \mathcal{C}_{m_2, n_2}^{m_2}$ with $m_1, m_2 \in \mathbb{N}$ be censoring plans and $X_{j:m_2:m_2}^{\mathcal{S}}, X_{i:m_1:m_1}^{\mathcal{R}}$ be progressively type II censored order statistics from the same absolutely continuous cdf F . If $1 \leq j \leq i$ and $\gamma_k(\mathcal{R}) \leq \gamma_k(\mathcal{S}), k = 1, \dots, j$, then $X_{j:m_2:m_2}^{\mathcal{S}} \leq_{lr} X_{i:m_1:m_1}^{\mathcal{R}}$.

Comparisons of vectors of progressively type II censored order statistics (generalized order statistics) with different cdfs and different censoring plans have been considered in [54]. In particular, they found the following property.

Theorem 9.2.6 Let $X_{i:m:n}^{\mathcal{R}}, Y_{i:m:n}^{\mathcal{R}}, 1 \leq i \leq m \leq n$, be progressively type II censored order statistics from absolutely continuous cdfs F and G , respectively, with $F \leq_{lr} G$ and censoring plan \mathcal{R} . Then, $X_{i:m:n}^{\mathcal{R}} \leq_{lr} Y_{i:m:n}^{\mathcal{R}}, 1 \leq i \leq m$.

Ordering of p -spacings is discussed in [54, 66–68].

Dispersive Order X is said to be smaller than Y in the dispersive order, i.e., $X \leq_{disp} Y$ or $F \leq_{disp} G$, iff $F^{\leftarrow}(x) - F^{\leftarrow}(y) \leq G^{\leftarrow}(x) - G^{\leftarrow}(y)$ for all $0 < y < x < 1$.

Results for the (multivariate) dispersive order are established in [55, 69–71]. For instance, [55] has shown a result as in Theorem 9.2.4 for the dispersive order provided that the cdf F has the DFR-property. Belzunce et al. [54] have shown that $X^{\mathcal{R}} \leq_{disp} Y^{\mathcal{R}}$ and that $X_{i:m:n}^{\mathcal{R}} \leq_{disp} Y_{i:m:n}^{\mathcal{R}}, 1 \leq i \leq m$, when $F \leq_{disp} G$.

Further orderings like mean residual life, total time on test, and excess wealth orders have also been discussed (see [72–75]). Results for the increasing convex order of generalized order statistics have been established in [76]. Orderings of residual life are discussed in [77, 78]. Stochastic orderings of INID progressively type II censored order statistics have been studied in [79].

Ageing Notions

Ageing properties have also been studied for progressively type II censored order statistics. For general references on ageing notions and their properties, we refer to, e.g., [80–82]. Results have been obtained for various ageing notions, e.g., increasing/decreasing failure rate (IFR/DFR), increasing/

decreasing failure rate on average (IFRA/DFRA), and new better/worse than used (NBU/NWU). Fundamental results on progressively type II censored order statistics for the most common ageing notions are mentioned subsequently.

IFR/DFR A cdf F is said to be IFR (DFR) iff the ratio $\frac{F(t+x)-F(t)}{1-F(t)}$ is increasing (decreasing) in $x \geq 0$ for all t with $F(t) < 1$.

If F exhibits a pdf then the IFR-/DFR-property means that the hazard rate function $\lambda_F = f/(1 - F)$ is increasing (decreasing).

IFRA/DFRA A cdf F is said to be IFRA (DFRA) iff, for the hazard function $R = -\log \bar{F}$, the ratio $R(x)/x$ is increasing (decreasing) in $x > 0$.

NBU/NWU A cdf F is said to be NBU (NWU) iff $\bar{F}(t+x) \leq (\geq) \bar{F}(t)\bar{F}(x)$ for all $x, t \geq 0$.

The IFR- and IFRA-property of progressively type II censored order statistics have been investigated in [33]. It has been shown that all progressively type II censored order statistics are IFR/IFRA provided that the baseline cdf F is IFR/IFRA. The respective result for the NBU-property as well as further results can be found in [83] in terms of sequential order statistics (see also [84]). Preservation properties are presented in [23], e.g., it has been proved that the DFR-property is preserved by spacings (see also [21]). The reversed hazard rate has been studied in [85]. Belzunce et al. [86] considered multivariate ageing properties in terms of nonhomogeneous birth processes and applied their results to generalized order statistics. A restriction to progressive censoring shows that progressively type II censored order statistics $X^{\mathcal{R}}$ are M-IFR if F is an IFR-cdf. Moreover, $X^{\mathcal{R}}$ is multivariate Polya frequency of order 2 (MPF₂) if the pdf of F is log-concave. Further notions of multivariate IFR/DFR and its applications to generalized order statistics have been discussed in [87]. The connection of ageing properties and residual life has been considered in [88] in terms of generalized order statistics.

Further Topics

The following probabilistic topics have also been discussed in progressive type II censoring, but, for brevity, we only mention them here briefly. Many publications deal with various kinds of characterizations of probability distributions [see, e.g., 1, Chapter 3.1]. Limit theorems have also been established imposing different assumptions on the censoring plans and distributions. For instance, [89] considered normal approximations using an approach inspired by Hoadley [90]. Cramer [91] discussed extreme value analysis which includes extreme, central, and intermediate progressively type

II censored order statistics [see also 92,93]. Counting process approaches in combination with limiting distributions have been extensively discussed in [94] [see also 95,96]. Hofmann et al. [97] has discussed a block censoring approach.

Information measures have also found some interest [see, e.g., 1, Chapter 9]. Results on the Fisher information have been established in, e.g., [98–102]. A detailed approach in terms of the more general model of generalized order statistics is discussed in [103, 104]. Asymptotic results are provided in [105]. Entropy-type measures are investigated in [106–110]. Kullback-Leibler-related measures are addressed in [106, 107, 111–115]. Pitman closeness for progressively type II censored data has been considered in, e.g., [116–119].

Concomitants for progressively type II censored order statistics have been addressed [120–122] (see also [1]).

As already mentioned above, progressive type II censoring has also been discussed under nonstandard conditions. Specifically, the underlying random variables X_1, \dots, X_n are supposed to be distributed according to some (multivariate) distribution function F^{X_1, \dots, X_n} . A general mixture representation of the distribution in terms of distributions of order statistics has been established by Fischer et al. [17]. Assuming independence but possibly different marginals, [16] found representations of the joint density functions. Inference in such a model has been discussed in [123]. Given a copula of the lifetimes X_1, \dots, X_n , [19, 20] addressed dependent random variables. Progressive censoring of random vectors has been discussed in [124] [see also 125, 126].

9.2.2 Inference

Inference for progressively type II censored data has been widely discussed in the literature. Most of the material is devoted to parametric inference. In the following, we present a selection of results for exponential and Weibull distribution. In addition, references to other distributions are provided. A standard reference for all these results is [1]. If nothing else is mentioned, we discuss inference on a single progressively type II censored sample $X_{1:m:n}, \dots, X_{m:m:n}$.

Point Estimation

The most popular parametric estimation concepts applied to progressively type II censored data are linear, likelihood, and Bayesian estimation. Assuming a location-scale family of distributions

$$\mathcal{F} = \left\{ F_{\mu, \vartheta} = F\left(\frac{\cdot - \mu}{\vartheta}\right) \mid \mu \in \mathbb{R}, \vartheta > 0 \right\},$$

with a known cdf F , a progressively type II censored sample $X^{\mathcal{R}}$ from $F_{\mu, \vartheta}$ can be written as a linear model:

$$X^{\mathcal{R}} = \mu \cdot \mathbf{1} + \vartheta Y^{\mathcal{R}} = \mu \cdot \mathbf{1} + \vartheta EY^{\mathcal{R}} + W^{\mathcal{R}} = B\theta + W^{\mathcal{R}},$$

where $EW^{\mathcal{R}} = 0$, $\text{Cov}(W^{\mathcal{R}}) = \vartheta^2 \Sigma$ denotes the variance-covariance matrix of $W^{\mathcal{R}}$, $\Sigma = \text{Cov}(Y^{\mathcal{R}})$, $B = [\mathbf{1}, \mathbf{b}]$ is the known design matrix, and $\theta = (\mu, \vartheta)'$ is the (unknown) parameter vector. Notice that the distribution of $Y^{\mathcal{R}}$ is parameter free, and it depends only on the standard member F .

Thus, as pointed out in [1, Chapter 11], least squares estimation can be applied in order to obtain the best linear unbiased estimator (BLUE) of θ (see, e.g., [127]) as

$$\hat{\theta} = (B' \Sigma^{-1} B)^{-1} B' \Sigma^{-1} X^{\mathcal{R}}.$$

Obviously, the estimator can be applied when the first and second moments of $X^{\mathcal{R}}$ can be computed (at least numerically). This has been done for many distributions. For instance, given exponential distributions, explicit expressions result since the respective moments have a closed form expression (see, e.g., [128–130]). For $m \geq 2$, the BLUEs $\hat{\mu}_{\text{LU}}$ and $\hat{\vartheta}_{\text{LU}}$ are given by

$$\begin{aligned} \hat{\mu}_{\text{LU}} &= X_{1:m:n} - \frac{\hat{\vartheta}_{\text{LU}}}{n}, \\ \hat{\vartheta}_{\text{LU}} &= \frac{1}{m-1} \sum_{j=2}^m (R_j + 1)(X_{j:m:n} - X_{1:m:n}). \end{aligned}$$

Further results for particular distributions are summarized in [1, Chapter 11]. In case of Weibull distributions, the model can be transformed to a linear model from extreme value distributions by a log-transformation of the data. Thus, estimators of the Weibull parameters can be obtained by using the BLUEs of the transformed parameters when the data results from an extreme value distribution. Results in this direction can be found in [24, 131, 132]. The mixture representation in terms of order statistics can also be utilized to compute the moments (see [17, 133, 134]).

Similar to least squares estimation, one can consider the best linear equivariant estimators (BLEEs). This problem has been discussed, e.g., by Balakrishnan et al. [135], Burkschat [136] (see also [137] for the best linear (risk) invariant estimators).

The most popular approach to the estimation problem is likelihood inference since the joint pdf given in (9.2) leads to tractable expressions in many situations (see [1, Chapter 12]). For generalized Pareto distributions, explicit expressions result. For exponentially distributed lifetimes with mean ϑ , the MLE is given by

$$\hat{\vartheta}_{\text{MLE}} = \frac{1}{m} \sum_{j=1}^m (R_j + 1) X_{j:m:n} = \frac{1}{m} \sum_{j=1}^m S_j^{\mathcal{R}}, \quad (9.8)$$

which is also the BLUE in this model. The representation in terms of the spacings $S_1^{\mathcal{R}}, \dots, S_m^{\mathcal{R}}$ is important in the analysis of the MLE since it enables easy derivation of

the exact distribution of the MLE. For two-parameter exponential distribution, explicit expressions for the MLEs are also available. For Weibull distribution $\text{Wei}(\vartheta, \beta)$, the MLE $(\hat{\vartheta}, \hat{\beta})$ of (ϑ, β) uniquely exists (see [138]). They are given by $\hat{\vartheta} = \frac{1}{m} \sum_{j=1}^m (R_j + 1) X_{j:m:n}^{\hat{\beta}}$ where, for the observed data $X_{j:m:n} = x_j$, $1 \leq j \leq m$, the estimate $\hat{\beta}$ is the unique solution of the equation:

$$\frac{m}{\beta} + \sum_{j=1}^m \log x_j - \frac{\sum_{j=1}^m (R_j + 1) \log(x_j) x_j^{\beta}}{\sum_{j=1}^m (R_j + 1) x_j^{\beta}} = 0.$$

The above equation has to be solved numerically, e.g., by the Newton-Raphson procedure. Ng et al. [98] proposed an EM-algorithm approach to compute the MLE (see also [139]) which, suitably adapted, has successfully been applied for other distributions, too. Results on likelihood inference for other distributions can be found in [1, Chapter 12]. Recent references for other distributions are, e.g., [140] (Rayleigh), [141] (modified Weibull), [142, 143] (Lindley), and [144] (Gompertz).

For some distributions, related concepts like modified and approximate maximum likelihood estimation have been discussed. The latter concept due to [145] has been successfully applied in many cases, e.g., for extreme value distribution [146] and Weibull distributions [147] (see also [1, Chapter 12.9.2]).

Bayesian inference has also been discussed considerably for progressively type II censored data under various loss functions (see [1, Chapter 15]). Under squared error loss function, the Bayes estimate of the scale parameter $\alpha = 1/\vartheta$ of an exponential lifetime is given by the posterior mean

$$\hat{\alpha}_{\text{B}} = \frac{a + m}{b + \sum_{j=1}^m (R_j + 1) X_{j:m:n}^{\beta}}$$

given a gamma prior

$$\pi_{a,b}(\alpha) = \frac{b^a}{\Gamma(a)} \alpha^{a-1} e^{-b\alpha}, \quad \alpha > 0,$$

with hyperparameters $a, b > 0$. Using a similar inverse gamma prior, [148] obtained the corresponding Bayes estimator of ϑ as

$$\hat{\vartheta}_{\text{B}} = \frac{1}{a + m - 1} \left(\sum_{j=1}^m \gamma_j (X_{j:m:n} - X_{j-1:m:n}) + b \right),$$

where $X_{0:m:n} = 0$. Two-parameter Weibull distribution with appropriate priors has been discussed in [149] and [150]. For further results, we refer to [1, Chapter 15].

Using a counting process approach, [94] and [151] have addressed nonparametric inference with progressively type

II censored lifetime data for the population cdf F and the survival function. For instance, they presented a Nelson-Aalen-type estimator and a smoothed hazard rate estimator as well as asymptotic results for these estimators. A survey is provided in [1, Chapter 20].

Statistical Intervals

Various kinds of statistical intervals have been discussed for progressively type II censored data. In particular, confidence intervals have been studied under different assumptions. In some situations, exact confidence intervals with level $1 - \alpha$ can be constructed using properties of the estimators. For exponential distribution, it follows from the independence of the spacings (9.3) that the distribution of the MLE $\hat{\vartheta}_{MLE}$ in (9.8) can be obtained as $2m\hat{\vartheta}_{MLE}/\vartheta \sim \chi^2(2m)$. Hence, $\left[\frac{2m\hat{\vartheta}_{MLE}}{\chi^2_{1-\alpha/2}(2m)}, \frac{2m\hat{\vartheta}_{MLE}}{\chi^2_{\alpha/2}(2m)} \right]$ is a two-sided $(1 - \alpha)$ -confidence interval for ϑ . Similarly, one may obtain confidence intervals and confidence regions for the two-parameter exponential distribution (see [1, Chapter 17], [152]). For Wei(ϑ, β)-distribution, [153] has obtained confidence intervals for the scale and shape parameters of a Wei(ϑ, β)-distribution using a transformation to exponential data and the independence of the spacings. An exact $(1 - \alpha)$ -confidence interval for β is given by

$$\mathcal{H} = \left[\psi^*(X^{\mathcal{R}}, F_{\alpha/2}(2(m-1), 2)), \psi^*(X^{\mathcal{R}}, F_{1-\alpha/2}(2(m-1), 2)) \right],$$

where $\psi^*(X^{\mathcal{R}}, \omega)$ is the unique solution for β of the equation

$$\sum_{j=2}^m (R_j + 1) \left(\frac{X_{j:m:n}}{X_{1:m:n}} \right)^\beta = \gamma_2 + n(m - 1)\omega.$$

Wang et al. [154] established a confidence interval for β using the pivotal quantity

$$\tau(X^{\mathcal{R}}, \beta) = 2 \sum_{j=1}^{m-1} \log \left(\frac{\sum_{i=1}^m (R_i + 1) X_{i:m:n}^\beta}{\sum_{i=1}^{j-1} (R_i + 1) X_{i:m:n}^\beta + \gamma_j X_{j:m:n}^\beta} \right).$$

They showed that an exact $(1 - \alpha)$ -confidence interval for β is given by

$$\mathcal{H} = \left[\tau^{-1}(X^{\mathcal{R}}, \chi^2_{\alpha/2}(2(m-1))), \tau^{-1}(X^{\mathcal{R}}, \chi^2_{1-\alpha/2}(2(m-1))) \right],$$

where $\tau^{-1}(X^{\mathcal{R}}, \omega)$ is the unique solution for β of the equation $\tau(X^{\mathcal{R}}, \beta) = \omega$ with $\omega > 0$. A simultaneous confidence region has been obtained by Wu [153]. The same ideas may be applied to Pareto distributions (see [155–158]). In [158, 159], and [160], optimal confidence regions are discussed (for a location-scale family, see [161]). Nonparametric confidence intervals for quantiles have been discussed in [133] (for

multiple samples, see [162, 163]). Exact confidence intervals based on conditional inference have been proposed for progressively type II censored data by Viveros and Balakrishnan [25] (see also [2, Chapter 9]). In particular, exponential, extreme value, log-gamma distributions, Pareto, and Laplace have been discussed. Asymptotic confidence intervals have been applied in various situations by assuming asymptotic normality of some pivotal quantities. The asymptotic variance is estimated by the observed likelihood. Generalized confidence intervals for distribution parameters using Weerahandi’s approach (see [164]) can be found in [154].

Furthermore, prediction intervals and tolerance intervals have been discussed. References for the latter concept are [134, 162, 165, 166]. The highest posterior density credible intervals have been established in [148, 149], and [167].

Prediction

Prediction problems have been discussed for both point and interval prediction, respectively (see [1, Chapter 16 & 17.4]). In particular, they have been considered for:

- (I) Progressively censored failure times at censoring steps $1, \dots, m$; in particular, the progressively censored ordered failure times $W_{j,1:R_j}, \dots, W_{j,R_j:R_j}$, $1 \leq j \leq m$, are predicted.
- (II) For future observations in the same sample (this is a particular case of (I) in the sense that the lifetimes of the items removed in the final progressive censoring step are predicted).
- (III) Observations of an independent future sample from the same population.

Problem (I) has been considered by Basak et al. [168] (the special case $W_{m,1:R_m}, \dots, W_{m,R_m:R_m}$ is addressed by Balakrishnan and Rao [169]). Given exponential lifetimes with unknown mean ϑ , they found that the best unbiased predictor of the r th ordered progressively censored lifetime $W_{j,r:R_j}$ in step j is given by

$$\hat{W}_{j,r:R_j}(X^{\mathcal{R}}) = X_{j:m:n} + \hat{\vartheta}_{MLE} \sum_{k=1}^r \frac{1}{R_j - k + 1},$$

where $\hat{\vartheta}_{MLE}$ is the MLE of ϑ . Further results have been obtained for extreme value distribution [168], normal distribution [170], and Pareto distribution [171]. Prediction intervals based on various prediction concepts (e.g., best linear unbiased prediction, maximum likelihood prediction, and median unbiased prediction) have been obtained for exponential and extreme value distributions in [168]. Normal and Pareto distributions are considered in [170] and [171], respectively. Generalized exponential and Rayleigh distributions are discussed in [172] and [173], respectively.



The best linear unbiased/equivariant prediction of future observations $X_{r+1:m:n}, \dots, X_{m:m:n}$, based on the first r progressively type II censored order statistics $X_{1:m:n}, \dots, X_{r:m:n}$, has been discussed in [136]. The same problem has been investigated in a Bayesian framework, too, and the corresponding results can be found in [148] and [152]. Prediction intervals for a general class of distributions, including exponential, Weibull, and Pareto distributions, can be seen in [174]. Results for Weibull distribution have also been presented in [175].

Problem (III) has mainly been discussed in a Bayesian framework. Relevant references are [141, 143, 175–183]. Results on nonparametric prediction of future observations can be found in [134, 162, 184].

Testing

Statistical tests under progressive censoring have mainly been discussed in the context of precedence-type testing, homogeneity, and goodness-of-fit tests. A good source for precedence-type tests is [185] (see also [1, Chapter 21]). Particular results can be found in [186] and [187]. Homogeneity tests based on several progressively type II censored samples have been addressed in [188]. For related results in terms of sequential order statistics, we refer to [23]. A review on goodness-of-fit-tests for exponential distributions, including a power study, has recently been presented in [114] (see also [1, Chapter 19]). Goodness-of-fit tests for location-scale families are discussed in, e.g., [189]. Tests have been constructed by means of spacings and deviations from the uniform distribution as well as from the empirical distribution function (e.g., Kolmogorov-Smirnov-type statistics; see [190, 191]). Furthermore, information measures like (cumulative) Kullback-Leibler information and entropy have been used (see [106, 111, 113]).

9.2.3 Experimental Design

Initiated by Balakrishnan and Aggarwala [2, Chapter 10], problems of experimental design have been discussed extensively for progressively type II censored lifetime experiments. A review on various results and optimality criteria has been provided by Balakrishnan and Cramer [1, Chapter 26]. Assuming that progressive type II censoring is carried out by design, the experimenter has to choose an appropriate censoring plan prior to the start of the experiment. Thus, assuming the sample size n and the number m of observed items as fixed, the censoring plan $\mathcal{R} = (R_1, \dots, R_m)$ has to be chosen in an optimal way. Burkschat [58] has formulated the problem as a mathematical optimization problem in a very general way (see also [192, 193]), that is, given a criterion $\psi : \mathcal{C}_{m,n}^m \rightarrow \mathbb{R}$, a censoring plan \mathcal{S} is said to be ψ -optimal if

$$\psi(\mathcal{S}) = \min_{\mathcal{R} \in \mathcal{C}_{m,n}^m} \psi(\mathcal{R}),$$

where $\mathcal{C}_{m,n}^m$ is given in (9.1). Various optimality criteria have been used, e.g., probabilistic criteria [58], variance criteria [2, 139, 192–194], information measures like Fisher information [100, 104, 105, 139] and entropy [107, 195, 196], optimal estimation of quantiles [139, 149, 197, 198], Pitman closeness [117], and optimal block censoring [97]. A detailed review is provided in [1, Chapter 26].

It turns out that the optimal designs depend heavily on both the optimality criterion to be used and the distributional assumption made. Due to the large number of admissible censoring plans, i.e., $\binom{n-1}{m-1}$ (see [1, p. 531]), [199] proposed a variable neighborhood search algorithm to identify optimal plans in a reasonable time. It should be mentioned that the so-called one-step censoring plans turn out to be optimal in many cases. This means that progressive censoring is carried out only at one failure time, whereas at the other failure times no censoring occurs. Such plans are discussed in [100, 102], and [200]. Recently, restrictions on censoring plans have been addressed in [49].

9.2.4 Connection of Progressive Type II Censoring to Coherent Systems

Cramer and Navarro [201] established a connection of failure data from coherent systems to progressively type II censored order statistics. They showed that the joint distribution of the component failures $(Y_{(1)}, \dots, Y_{(m)})$ (given the number $M = m$ of component failures leading to the system failure) in a coherent system can be seen as a mixture of progressively type II censored order statistics:

$$P^{Y_{(1)}, \dots, Y_{(m)} | M=m} = \sum_{\mathbf{r} \in \mathcal{S}_m} P(\mathcal{R} = \mathbf{r} | M = m) P^{X_{1:m:n}^{\mathbf{r}}, \dots, X_{m:m:n}^{\mathbf{r}}}, \quad (9.9)$$

where \mathcal{S}_m denotes the set of all admissible censoring plans $\mathbf{r} = (r_1, \dots, r_m)$ of length m . The probabilities $P(\mathcal{R} = \mathbf{r} | M = m)$ depend only on the structure of the coherent system and therefore can be calculated directly [see also 202]. Utilizing this connection, inference for coherent system data can be carried out using inferential methods for progressively type II censored data. For exponentially distributed lifetimes as well as PHR models, we refer to [201], while Weibull distribution is discussed in [203].

Cramer and Navarro [202] applied this connection to define a progressive censoring signature (PC-signature) which can be used to compare the lifetimes of different coherent systems with respect to stochastic orderings.

9.2.5 Connection of Progressive Type II Censoring to Ordered Pooled Samples

In (9.9), a mixture of some random variables in terms of progressively type II censored order statistics has been established. A similar mixture representation has been found in the context of pooling two independent type II censored samples. Let $X_{1:m}, \dots, X_{r:n}$ and $Y_{1:m}, \dots, Y_{s:m}$ be independent right censored samples from a uniform distribution with sample sizes n and m , respectively. Without loss of any generality, let $r \geq s$ and denote the ordered pooled sample by $W_{(1)} \leq \dots \leq W_{(r+s)}$. Then, [162] showed that the joint distribution of the ordered pooled sample $W_{(1)}, \dots, W_{(r+s)}$ is a mixture of uniform progressively type II censored order statistics, that is,

$$P^{W_{(1)}, \dots, W_{(r+s)}} = \sum_{j=0}^{r-1} \pi_j P^{(T_{1:r+s:n+m}^{\mathcal{R}_j}, \dots, T_{r+s:r+s:n+m}^{\mathcal{R}_j})} + \sum_{j=0}^{s-1} \pi_j^* P^{(T_{1:r+s:n+m}^{\mathcal{R}_j^*}, \dots, T_{r+s:r+s:n+m}^{\mathcal{R}_j^*})},$$

with appropriately chosen discrete probability distributions π_0, \dots, π_{r-1} and $\pi_0^*, \dots, \pi_{s-1}^*$ and two-step censoring plans $\mathcal{R}_j, 0 \leq j \leq r-1$ and $\mathcal{R}_j^*, 0 \leq j \leq s-1$, respectively [162], [1, Section 17.1.6]. An extension to multiple pooled samples is presented in [204].

9.3 Progressive Type I Censoring

As mentioned in Sect. 9.1, progressive type I censoring as introduced in [5] does not have a prefixed termination time, that is, the last censoring time T_m (see Fig. 9.1) has not been considered as termination time. As pointed out in [6], inference for this model has been considered up to the early 1990s (see, e.g., [5, 205–209], and the monographs by Nelson [210] and Cohen and Whitten [211]). In the following, we call this censoring scheme Cohen’s *progressive censoring with fixed censoring times*. The understanding of T_m as a time censoring point seems to have changed after the publication of the monograph [2] [see also 7]. Since then, almost all publications dealing with progressive type I censoring interpret T_m as the termination time of the experiment.

9.3.1 Distributional Results for Cohen’s Progressive Censoring with Fixed Censoring Times

We start with a short review of progressive censoring with fixed censoring times as presented in [6]. In principle, the

procedure is quite similar to progressive type II censoring, but the censoring times are fixed in advance. Due to this property, we have to distinguish the initially planned censoring plan $\mathcal{R}^0 = (R_1^0, \dots, R_m^0)$ and the effectively applied censoring plan $\mathcal{R} = (R_1, \dots, R_m)$ (see [1, 2]) where

$$\mathcal{R}^0 = (R_1^0, \dots, R_m^0) \in \mathcal{C}_{l,n}^m = \left\{ (r_1, \dots, r_m) \in \mathbb{N}_0^m \mid r_{\bullet m} \leq n - l \right\}, \quad l \in \{0, \dots, n\},$$

with $m, n \in \mathbb{N}$ and $0 \leq R_j \leq R_j^0, 1 \leq j \leq m$, and $\mathcal{C}_{l,n}^m$ denotes the set of admissible censoring plans. As mentioned in the introduction, the censoring times of a progressively type I censored life test are prefixed, and the number of observations is random. Thus, we get a sample $X_{1:K:n}^l \leq \dots \leq X_{K:K:n}^l$ with random censoring plan \mathcal{R} and random sample size K . Notice that the important difference between progressive censoring with fixed censoring times and progressive type I censoring is the fact that we can ensure a minimum number of observations, that is, $K \geq n - R_{\bullet m}^0 \geq n - (n - l) = l$. Moreover, it is possible to observe values exceeding the threshold T_m . Progressive type I censoring can be interpreted as a type I hybrid version of progressive censoring with fixed censoring times (see Sect. 9.4.2). Denoting the number of observations in the intervals

$$(-\infty, T_1], (T_1, T_2], \dots, (T_{m-1}, T_m], (T_m, \infty)$$

by the random variables $D_1, D_2, \dots, D_m, D_{m+1}$ and by d_1, \dots, d_{m+1} their realizations, the effectively applied censoring numbers are given by

$$R_j = R_j(d_j) = \min \{ R_j^0, [n - d_{\bullet j} - R_{\bullet j-1}]_+ \}, \quad 1 \leq j \leq m,$$

where $n - d_{\bullet i} - R_{\bullet i-1}$ equals the number of units still remaining in the experiment before the i th withdrawal at time T_i . Notice that D_{m+1} is a (deterministic) function of D_1, \dots, D_m , i.e., $D_{m+1} = n - D_{\bullet m} - R_{\bullet m}$. Then, the set

$$\mathcal{D}_{(m+1)} = \{ a_{m+1} \in \mathbb{N}_0^{m+1} \mid a_i \leq [n - a_{\bullet i-1} - R_{\bullet i-1}(a_{i-1})]_+, \quad i = 1, \dots, m, \quad a_{m+1} = [n - a_{\bullet m} - R_{\bullet m}(a_m)]_+ \}$$

denotes the support of (D_1, \dots, D_{m+1}) for a progressively censored life test with fixed censoring times.

Similarly to Theorem 9.2.1, we get the following quantile representation.

Theorem 9.3.1 *Suppose $X_{1:K:n}^l \leq \dots \leq X_{K:K:n}^l$ and $U_{1:K:n}^l \leq \dots \leq U_{K:K:n}^l$ are progressively censored order statistics with fixed censoring times based on a continuous cdf F and a standard uniform distribution, respectively. The censoring times*

are given by T_1, \dots, T_m and $F(T_1), \dots, F(T_m)$, respectively. Then,

$$\left(X_{1:K:n}^I, \dots, X_{K:K:n}^I \right) \stackrel{d}{=} \left(F^{\leftarrow}(U_{1:K:n}^I), \dots, F^{\leftarrow}(U_{K:K:n}^I) \right).$$

Assuming $X_1, \dots, X_n \stackrel{iid}{\sim} F$ with an absolutely continuous cdf F and a pdf f , $T_{m+1} = \infty$, and $R_{m+1} = 0$, the joint pdf $f^{X^I, \mathbf{D}_{m+1}}$ of $\mathbf{X}^{I, \mathcal{R}} = (X_{1:K:n}^I, \dots, X_{K:K:n}^I)$ and $\mathbf{D}_{m+1} = (D_1, \dots, D_{m+1})$ is given by

$$\begin{aligned} f^{X^I, \mathcal{R}, \mathbf{D}_{m+1}}(\mathbf{x}_k, \mathbf{d}_{m+1}) &= \left\{ \prod_{j=1}^{m+1} \binom{n - d_{\bullet j-1} - R_{\bullet j-1}}{d_j} d_j! [1 - F(T_j)]^{R_j} \right\} \left\{ \prod_{i=1}^k f(x_i) \right\} \end{aligned} \tag{9.10}$$

for $\mathbf{d}_{m+1} \in \mathcal{D}_{(m+1)}$ with $k = d_{\bullet m+1}$ and $x_1 \leq \dots \leq x_k$. Clearly, (9.10) can be rewritten as

$$f^{X^I, \mathcal{R}, \mathbf{D}_{m+1}}(\mathbf{x}_k, \mathbf{d}_{m+1}) = C(\mathbf{d}_{m+1}) \prod_{j=1}^{m+1} [1 - F(T_j)]^{R_j} \prod_{i=1}^k f(x_i)$$

which illustrates the structural similarities to the pdf under progressive type II censoring given in (9.2). Notice that the value of $\mathbf{D}_{(m+1)}$ is defined uniquely by $\mathbf{X}^{I, \mathcal{R}} = (X_{1:K:n}^I, \dots, X_{K:K:n}^I)$.

The joint probability mass function $p^{\mathbf{D}_{m+1}}$ of \mathbf{D}_{m+1} is given by

$$\begin{aligned} p^{\mathbf{D}_{m+1}}(\mathbf{d}_{m+1}) &= \prod_{j=1}^{m+1} \binom{n - d_{\bullet j-1} - R_{\bullet j-1}}{d_j} [F(T_j) - F(T_{j-1})]^{d_j} \\ &\times [1 - F(T_j)]^{R_j}, \mathbf{d}_{m+1} \in \mathcal{D}_{(m+1)}. \end{aligned}$$

Further, the conditional density function of $\mathbf{X}^{I, \mathcal{R}}$, given \mathbf{D}_{m+1} , is given by

$$f^{X^I, \mathcal{R} | \mathbf{D}_{m+1}}(\mathbf{x}_k | \mathbf{d}_{m+1}) = \prod_{j=1, d_j > 0}^{m+1} f_{1 \dots d_j; d_j}^{(j)}(\mathbf{x}_{d_j}) \mathbb{I}_{(T_{j-1}, T_j]}(\mathbf{x}_{d_j}) \tag{9.11}$$

for $\mathbf{d}_{m+1} \in \mathcal{D}_{(m+1)}$ with $m = d_{\bullet m+1}$ and $x_1 \leq \dots \leq x_k \cdot f_{1 \dots d_j; d_j}^{(j)}$ denotes the density function of order statistics $X_{1:d_j}^{(j)}, \dots, X_{d_j:d_j}^{(j)}$ from the (doubly) truncated cdf F in the interval $(T_{j-1}, T_j]$.

As for progressive type II censoring, the conditional block independence of progressively censored order statistics with fixed censoring times holds which for progressive type I censoring and progressive type II censoring has first been established by Iliopoulos and Balakrishnan [39]. It follows directly from the joint pdf. Conditionally on $(D_1, \dots, D_{m+1}) = (d_1, \dots, d_{m+1})$, the progressively censored order statistics with fixed censoring times are block independent, that is, the random vectors

$$\left(X_{d_{\bullet j-1}+1:d_{\bullet m^*+1}:n}^I, \dots, X_{d_{\bullet j}:d_{\bullet m^*+1}:n}^I \right), \quad j \in \{1 \leq i \leq m+1 \mid d_i > 0\},$$

are independent with

$$\begin{aligned} &\left(X_{d_{\bullet j-1}+1:d_{\bullet m^*+1}:n}^I, \dots, X_{d_{\bullet j}:d_{\bullet m^*+1}:n}^I \right) \\ &\stackrel{d}{=} \left(X_{1:d_j}^{(j)}, \dots, X_{d_j:d_j}^{(j)} \right), \quad j \in \{1 \leq i \leq m+1 \mid d_i > 0\}, \end{aligned}$$

where $X_{1:d_j}^{(j)}, \dots, X_{d_j:d_j}^{(j)}$ are order statistics from the (doubly) truncated cdf F in the interval $(T_{j-1}, T_j]$, $j \in \{1 \leq i \leq k \mid d_i > 0\}$.

9.3.2 Distributional Results for Progressive Type I Censoring

Since progressive type I censoring results from Cohen's progressive censoring model with fixed censoring times by interpreting the final censoring time T_m as a threshold or termination point, the respective sample results from $X_{1:K:n}^I \leq \dots \leq X_{K:K:n}^I$ is given by $\mathbf{X}^{I, \mathcal{R}, T_m} = (X_{1:K:n}^I, \dots, X_{D_{\bullet m}:K:n}^I)$. Notice that this sample may result in an empty sample when all items are either progressively censored at T_1, \dots, T_{m-1} or right censored at T_m . Thus, no failures have been observed due to the time censoring at T_m . As a consequence, inferential results are often obtained and discussed subject to the assumption that at least one failure has been observed, that is, $D_{\bullet m} \geq 1$ which happens with probability

$$P(D_{\bullet m} \geq 1) = 1 - \prod_{i=1}^m (1 - F(T_i))^{R_i^0}.$$

Then, the distributional results presented before can be applied to progressive type I censoring. For instance, the quantile representation in Theorem 9.3.1 holds, too. The pdf of $\mathbf{X}^{I, \mathcal{R}, T_m}$ and \mathbf{D}_m can be seen as a marginal pdf of (9.10) with an appropriate restriction on the domain. This leads to the pdf

$$f^{X^I, \mathcal{R}, T_m, D_m}(x_k, d_m) = \left\{ \prod_{i=1}^m \binom{n - d_{\bullet i-1} - R_{\bullet i-1}}{d_i} d_i! [1 - F(T_i)]^{R_i} \right\} \left\{ \prod_{j=1}^k f(x_j) \right\} \quad (9.12)$$

for $d_m \in \mathcal{D}_{(m)}$ with $k = d_{\bullet m} \geq 1$ and $x_1 \leq \dots \leq x_k \leq T_m$ (see [1, p. 121]). Notice that R_m is defined differently in comparison with (9.10).

Apart from the above presented results, almost no probabilistic results seem to be available for (Cohen’s) progressively type I censored order statistics. Obviously, this is caused by the problems due to the random sample size K and the random censoring plan \mathcal{R} . Nevertheless, numerous inferential results have been obtained.

9.3.3 Inference

Most of the results established in progressive type I censoring are connected to likelihood inference. Since the likelihood functions are given by the joint pdfs (9.10) and (9.12), respectively, the MLEs can be obtained by direct optimization. In the following, we present only the progressive type I censoring model (with time censoring). Similar results can be obtained for Cohen’s progressive censoring with fixed censoring times model (see, e.g., [6]). A summary with more details is provided in [1, Chapter 12]. Notice that, due to the similarity of the likelihood function to the progressive type II censoring case, the computation of MLEs proceeds quite similarly to this case. In particular, explicit expressions result in the same cases, and the likelihood equations are similar (replace the censoring time T_i in progressive type I censoring by the observed failure time $X_{i:m:n}$ in progressive type II censoring; cf. (9.8) and (9.13)). For exponentially distributed lifetimes with mean ϑ , one gets the MLE as

$$\widehat{\vartheta}_{MLE}^I = \frac{1}{K} \left[\sum_{i=1}^K X_{i:K:n}^I + \sum_{j=1}^m R_j T_j \right]. \quad (9.13)$$

Although the structure is similar to the MLE under progressive type II censoring, the distribution of the estimator is quite complicated. Using a moment generating function approach, [212] established the conditional pdf of $\widehat{\vartheta}_{MLE}^I$, given $D_{\bullet m} \geq 1$, as a generalized mixture of (shifted) gamma distributions (for a direct approach under progressive censoring with fixed censoring times, see [6]). Establishing the stochastic monotonicity of the (conditional) survival function by the three monotonicity lemmas (see also [212–214]), constructed exact (conditional) confidence intervals for ϑ using the method of pivoting the cdf (see [215–217]). A multi-

sample model has been studied in [218] who presented an alternate representation of the pdf in terms of B-spline functions. Two-parameter exponential distribution $\text{Exp}(\mu, \vartheta)$ has been considered, e.g., in [211, 219], and [220]. Cohen [219] proposed also modified MLEs. Weibull distribution has been discussed in [205, 219, 221, 222], and [2]. Explicit expressions for the MLEs are not available, and the estimates have to be computed by numerical procedures. Balakrishnan and Kateri [138] have established the existence and uniqueness of the MLEs. Three-parameter Weibull distributions are considered in [206, 208], and [223]. Further distributions considered are, e.g., extreme value distribution [219], normal distribution [5], Burr-XII distribution [209], and logistic distribution [224].

9.4 Sampling Models Based on Progressively Censored Data

Progressively type I and type II censored data have been considered as a basis for inferential purposes in various models. In the following, we sketch some of these applications and provide some recent references.

9.4.1 Progressive (Type I) Interval Censoring

In progressive type I censoring, the number of observations observed between censoring times is random. It is assumed that only these numbers are observed (see Fig. 9.5), whereas the exact values of the failure times are not observed. This kind of data has been introduced in [7] (see also [1, Chapter 12]) assuming an absolutely continuous cdf F_θ . This yields the likelihood function (cf. (9.11))

$$L(\theta) \propto \prod_{j=1}^m (F_\theta(T_j) - F_\theta(T_{j-1}))^{d_j} \overline{F}_\theta^{R_j}(T_j),$$

where $\theta = (\theta_1, \dots, \theta_p)' \in \Theta \subseteq \mathbb{R}^p$ denotes the parameter vector and d_1, \dots, d_m are realizations of the number of observed failures D_1, \dots, D_m . $T_0 = -\infty < T_1 < \dots < T_m$ are the censoring times, and $\mathcal{R} = (R_1, \dots, R_m)$ is the effectively applied censoring plan.

Inferential results have been established for various distributions. Asymptotic results for MLEs with general distribution have been established in [225]. Exponential distribution has been discussed in [7] and [226]. For (inverse) Weibull distribution, we refer to [226, 227], and [228]. Further distributions considered are generalized exponential distributions [228–230], generalized Rayleigh distribution [231], gamma distribution [232], and Burr-XII [233]. Further examples are given in [1, Chapter 18].

Under progressive type I censoring, the optimal choice of both inspection times and censoring proportions has been addressed by many authors [1, Chapter 18.2 & 18.3]. Other relevant references in this direction are [225, 233–241].

9.4.2 Progressive Hybrid Censoring

In progressive hybrid censoring, progressive type II censored data is subject to, e.g., additional time censoring at some threshold T . There are many variations available in the literature so far. For reviews, see [242], [1, Chapter 5 & 14], [12, 243, 244] and the recent monograph [380]. For illustration, we sketch the idea of the two basic hybrid censoring models under progressive censoring. Let $D = \sum_{i=1}^m \mathbb{I}_{(-\infty, T]}(X_{i:m:n})$ denote the total number of observed failures. As in [10], we perceive the data with possibly less than m observed failure times as a sample of size m by adding the censoring time in the required number. For a progressively type II censored sample $X_{1:m:n}, \dots, X_{m:m:n}$ with censoring plan \mathcal{R} , type I progressively hybrid censored order statistics $X_{1:m:n}^{h,I}, \dots, X_{m:m:n}^{h,I}$ are defined via

$$X_{j:m:n}^{h,I} = \min(X_{j:m:n}, T), \quad 1 \leq j \leq m. \tag{9.14}$$

Notice that the names *type I/type II* progressive hybrid censoring are differently used in the literature which may result in some confusion. From (9.14), it is evident that the sample may include both observed failure times and censoring times. Conditionally on $D = d, d \in \{0, \dots, m\}$, we have

$$X_{1:m:n}^{h,I}, \dots, X_{m:m:n}^{h,I} | (D = d) \stackrel{d}{=} X_{1:m:n}, \dots, X_{d:m:n}, \underbrace{T, \dots, T}_{m-d \text{ times}}.$$

For $d = 0$, the experiment has been terminated before observing the first failure, and, thus, the sample is given by the constant data $(T, \dots, T) \in \mathbb{R}^m$. Some probabilistic results have been obtained (see [1, Chapter 5], [10]). For instance, as under progressive type II censoring, a quantile representation similar to that given in Theorem 9.2.1 holds. The (conditional) joint pdf is given by Childs et al. [see also 8, 9]:

$$f_{j:m:n}^{h,I, 1 \leq j \leq m | D=d}(t_d, T^{*(m-d)}) = \frac{C_d}{P(D=d)} (1 - F(T))^{\gamma_{d+1}} f_{1, \dots, d; d:n-\gamma_{d+1}}^{\mathcal{R}_d}(t_d), \quad t_1 \leq \dots \leq t_d \leq T,$$

where $f_{1, \dots, d; d:n-\gamma_{d+1}}^{\mathcal{R}_d}$ denotes the pdf of the progressively type II censored sample with a censoring plan \mathcal{R}_d . In case of the exponential distribution, [10] established the (conditional) joint density function of the spacings $W_{j:m:n}^{h,I} = \gamma_j(X_{j:m:n}^{h,I} - X_{j-1:m:n}^{h,I}), 1 \leq j \leq d$, as

$$f_{j:m:n}^{h,I, 1 \leq j \leq d | D=d}(w_d) = \frac{\gamma_{d+1} e^{-\gamma_{d+1}(T-\mu)/\vartheta}}{\vartheta f_{d+1:m:n}(T)} \times \left[\prod_{j=1}^d \frac{1}{\vartheta} \exp \left\{ - \left(1 - \frac{\gamma_{d+1}}{\gamma_j} \right) \frac{w_j}{\vartheta} \right\} \right],$$

$$w_d \in \mathcal{W}_d(T),$$

with support

$$\mathcal{W}_d(T) = \left\{ w_d \mid w_j \geq 0, 1 \leq j \leq d, \sum_{j=1}^d \frac{w_j}{\gamma_j} \leq T - \mu \right\}.$$

As a difference to the case of progressive type II censoring, the spacings are no longer independent although the pdf exhibits a product structure. These results can be utilized to obtain the exact distribution of the MLE for exponential lifetimes. A moment generating function approach is advocated in, e.g., [8]. The MLE of ϑ exists provided $D > 0$ and is given by

$$\hat{\vartheta} = \frac{1}{D} \left[\sum_{j=1}^D \gamma_j (X_{j:m:n}^{h,I} - X_{j-1:m:n}^{h,I}) + \gamma D_{d+1} T \right].$$

Its distribution can be written in terms of B-spline functions (see [10]) or in terms of shifted gamma functions (see [8]). The connection of the particular representations has been studied in [243]. The result can be applied to construct exact (conditional) confidence intervals by pivoting the cdf since the corresponding survival function is stochastically monotone (see [213, 214, 245]). For the multi-sample case, we refer to [246]. Results for two-parameter exponential distribution are given in [10] and [247]. Inference for Weibull distribution has been discussed in [248]. Results for other distributions can be found in, e.g., [249–255]. Optimal censoring plans are discussed in [256, 257].

Childs et al. [8] and Kundu and Joarder [9] proposed an alternative hybrid censoring procedure called type II progressive hybrid censoring. Given a (fixed) threshold time T , the life test terminates at $T_2^* = \max\{X_{m:m:n}, T\}$. This approach guarantees that the life test yields at least the observation of m failure times. Given the progressively type II censored sample $X_{1:m:n}, \dots, X_{m:m:n}$ with an initially planned censoring plan $\mathcal{R} = (R_1, \dots, R_m)$, the right censoring at time $X_{m:m:n}$ is not carried out. The monitoring of the failure times after $X_{m:m:n}$ is continued until time T is reached or the maximum in the extended progressively type II censored sample

$$X_{1:m+R_m:n}, \dots, X_{m:m+R_m:n}, X_{m+1:m+R_m:n}, \dots, X_{m+R_m:m+R_m:n} \tag{9.15}$$

is observed. Notice that this sample can be viewed as progressively type II censored data with extended censoring plan $\mathcal{R}^* = (R_1, \dots, R_{m-1}, 0, \dots, 0)$ of length $m + R_m$. Furthermore, $\gamma_j = \sum_{i=j}^m (R_i + 1)$, $j = 1, \dots, m - 1$, $\gamma_j = m + R_m - j + 1$, $j = m, \dots, m + R_m$. As in the case of type I progressive hybrid censoring, the random counter $D = \sum_{i=1}^{m+R_m} \mathbb{I}_{(-\infty, T]}(X_{i:m+R_m:n})$ represents the sample size having support $\{0, \dots, m + R_m\}$. Again, the exact distribution of the MLE given by

$$\widehat{\vartheta} = \begin{cases} \frac{1}{m} \sum_{j=1}^m (R_j + 1) X_{j:m:n}, & D < m \\ \frac{1}{D} \left(\sum_{j=1}^m (R_j + 1) X_{j:m:n} + \sum_{j=m+1}^D X_{j:m+R_m:n} + \gamma_{D+1} T \right), & D \geq m \end{cases}$$

can be obtained. Furthermore, exact confidence intervals can be established (see [8, 258]) since the cdf is stochastically monotone (see [214]). For the multi-sample case, we refer to [259]. Inferential results for this kind of data have been obtained in, e.g., [8, 9, 258, 260]. Mokhtari et al. [261], Alma and Arabi Belaghi [262], and Noori Asl et al. [263].

There are many extensions on this basic progressive hybrid censoring. Generalized progressive hybrid censoring is discussed in, e.g., [243, 264–270]. Further extensions can be found in [12] and [271]. The Fisher information in hybrid censoring schemes is discussed in [272] and [273]. Furthermore, interval censored data have been studied in [274].

9.4.3 Adaptive Progressive Censoring

A common feature of the abovementioned progressive censoring schemes is that the design of the experiment (i.e., initially planned censoring plan, censoring times) is prefixed, that is, these quantities are known in advance. Since such a design may not be possible or be useful in practical situations, [13] came up with the idea that the censoring plan may be adapted during the experiment. Given some prefixed censoring plan $\mathcal{S} = (S_1, \dots, S_m)$ and a threshold T , the plan is adapted after step $j^* = \max\{j : X_{j:m:n} < T\}$ such that no further censoring is carried out until the m th failure time has been observed. Hence, the censoring plan is changed at the progressive censoring step $j^* + 1$, i.e., at the first observed failure time exceeding the threshold T . The effectively applied censoring plan in the Ng-Kundu-Chan model is given by $\mathcal{S}^* = (S_1, \dots, S_{j^*}, 0, \dots, 0, n - m - \sum_{i=1}^{j^*} S_i)$. This model has been extensively investigated, and many results have been obtained (see [248, 260, 275, 276]).

A general approach to adaptive progressive censoring has been proposed by Cramer and Iliopoulos [15] allowing for a flexible choice of the censoring plan and the censoring times. This approach covers both adaptive progressive type I and adaptive progressive type II censoring. Adaptive progressive

type II censoring has been discussed in detail in [14] who particularly showed that the model covers the Ng-Kundu-Chan model as well as the model of progressive type II censoring with random removals (see also [1, Chapter 6]). The latter model has been proposed by Yuen and Tse [277] assuming that the censoring numbers are chosen according to some probability distribution on the set of possible censoring numbers. Further references discussing this model are, e.g., [278–280]. For interval censored data, we refer to [281] and [282]. Flexible progressive censoring introduced in [283] can also be seen as a special adaptive progressive censoring model (see also [284, 285]).

9.4.4 Reliability and Stress-Strength Reliability

Applications in reliability based on progressively type II censored data have been addressed by many authors. The analysis is mostly based on a single progressively type II censored sample $X_{1:m:n}, \dots, X_{m:m:n}$, but the situation of multiple samples has also been taken into account. In the following, we summarize some scenarios where this kind of data has been considered.

Given a lifetime X with cdf F , the reliability function $R = \bar{F} = 1 - F$ can be estimated parametrically and nonparametrically. Nonparametric estimators under progressive censoring are mentioned in Sect. 9.2.2. Parametric estimators of R_θ can be constructed as plug-in estimators by replacing θ by an appropriate estimator $\widehat{\theta}$, e.g., the MLE (see, e.g., [142, 286–291]). Furthermore, Bayesian approaches have also been extensively discussed. However, for exponential distributions and $t \in \mathbb{R}$, the UMVUE of $R_\theta(t) = P_\theta(X > t)$ is given by (see (9.8))

$$\widehat{R}(t) = \left(1 - \frac{t}{m\widehat{\vartheta}_{\text{MLE}}} \right)^{m-1} \mathbb{I}_{[t, \infty)}(m\widehat{\vartheta}_{\text{MLE}})$$

(see [292]). The result can be slightly extended to an exponential family with cdf F_ϑ defined by $F_\vartheta(t) = 1 - \exp(-g(t)/\vartheta)$ and a suitable function g (see, e.g., [23, 293]).

Inference for the stress-strength reliability $R = P(X < Y)$ has also been addressed under progressive censoring for various distributions (for a general account, see [294]). For exponential distribution, the problem has been considered in terms of Weibull exponential distributions in [295] and [296]. For two independent progressively type II censored samples $X_{1:m:n}^{\mathcal{R}}, \dots, X_{m:m:n}^{\mathcal{R}}$ and $Y_{1:r:s}^{\mathcal{S}}, \dots, Y_{r:r:s}^{\mathcal{S}}$, based on exponential distributions $\text{Exp}(\vartheta_1)$ and $\text{Exp}(\vartheta_2)$, the MLEs of the parameters are given by $\widehat{\vartheta}_{j,\text{MLE}}$, $j = 1, 2$, as in (9.8). Then, the MLE of $R = \frac{\vartheta_2}{\vartheta_1 + \vartheta_2}$ is given by $R_{\text{MLE}} = \frac{\widehat{\vartheta}_2}{\widehat{\vartheta}_1 + \widehat{\vartheta}_2}$. Furthermore, the UMVUE is given by

$$\widehat{R}_{UMVUE} = \begin{cases} \sum_{j=0}^{r-1} (-1)^j \binom{r-1}{m+j-1} \left(\frac{m\widehat{\vartheta}_{1,MLE}}{r\widehat{\vartheta}_{2,MLE}} \right)^j, & m\widehat{\vartheta}_{1,MLE} \leq r\widehat{\vartheta}_{2,MLE} \\ 1 - \sum_{j=0}^{m-1} (-1)^j \binom{m-1}{r+j-1} \left(\frac{r\widehat{\vartheta}_{2,MLE}}{m\widehat{\vartheta}_{1,MLE}} \right)^j, & m\widehat{\vartheta}_{1,MLE} > r\widehat{\vartheta}_{2,MLE} \end{cases}$$

$$f^{X^{\mathcal{R}}, C}(x_m, c_m) = \left(\prod_{j=1}^m \gamma_j \right) \prod_{i=1}^m \left[[f_1(x_i)\overline{F}_2(x_i)]^{\mathbb{I}_{\{1\}}(c_i)} [f_2(x_i)\overline{F}_1(x_i)]^{\mathbb{I}_{\{2\}}(c_i)} [\overline{F}_1(x_i)\overline{F}_2(x_i)]^{R_i} \right].$$

Saraçoğlu et al. [see also 297, who addressed Bayesian inference, too]. Confidence intervals are discussed in [295, 296], and [297]. Two-parameter exponential distributions with common location parameter are investigated in [295, 296] [see also 1, Chapter 24]. Other distributions considered in the literature are generalized (inverted) exponential distribution [289, 298], Weibull distribution [299, 300], generalized Pareto distributions [301], the PHR model [302, 303], Birnbaum-Saunders distribution [304], generalized logistic distribution [305], and finite mixtures [306].

Stress-strength models under joint progressive censoring have been considered in [307]. Progressively type I interval censored data has been discussed by Bai et al. [306].

9.4.5 Competing Risks

In competing risk modeling, it is assumed that a unit may fail due to several causes of failure. For two competing risks, the lifetime of the i th unit is given by

$$X_i = \min \{X_{1i}, X_{2i}\}, \quad i = 1, \dots, n,$$

where X_{ji} denotes the latent failure time of the i th unit under the j th cause of failure, $j = 1, 2$. In most models considering competing risks under a progressive censoring scheme, the latent failure times are assumed to be independent with $X_{ji} \sim F_j, j = 1, 2, i = 1, \dots, m$. Additionally, the sample X_1, \dots, X_n is progressively type II censored, and it is assumed that the cause of each failure is known. Therefore, the available data are given by

$$(X_{1:m:n}, C_1), (X_{2:m:n}, C_2), \dots, (X_{m:m:n}, C_m),$$

where $C_i = 1$ if the i th failure is due to first cause and $C_i = 2$ otherwise. The observed data is denoted by $(x_1, c_1), (x_2, c_2), \dots, (x_m, c_m)$. Further, we define the indicators

$$\mathbb{I}_{\{j\}}(C_i) = \begin{cases} 1, & C_i = j \\ 0, & \text{otherwise} \end{cases}.$$

Thus, the random variables $m_1 = \sum_{i=1}^m \mathbb{I}_{\{1\}}(C_i)$ and $m_2 = \sum_{i=1}^m \mathbb{I}_{\{2\}}(C_i)$ describe the number of failures due to the first and the second cause of failure, respectively. Given the assumptions, m_1 and m_2 are binomials with sample size m and probability of success $R = P(X_{11} \leq X_{21})$ and $1 - R$, respectively. For a given censoring plan $\mathcal{R} = (R_1, \dots, R_m)$, the joint pdf is given by Kundu et al. [see 308]

[308] discussed competing risks for $\text{Exp}(\vartheta_j)$ -distributions, $j = 1, 2$, under progressive type II censoring. The MLEs of the parameters are given by

$$\widehat{\vartheta}_j = \frac{1}{m_j} \sum_{i=1}^m (R_i + 1)X_{i:m:n}, \quad j = 1, 2,$$

provided that $m_j > 0$. In this framework, inferential topics like point and interval estimation as well as prediction problems have been discussed. To keep things short, we provide references for further reading. Competing risks under progressive type II censoring have been considered for, e.g., Weibull distribution [309–311], Lomax distribution [312], half-logistic distribution [313], and Kumaraswamy distribution [314].

Progressive type I interval censoring in the presence of competing risks has been investigated by Wu et al. [234], Azizi et al. [315], and Ahmadi et al. [316]. Competing risks under hybrid censoring are investigated in, e.g., [317–325].

9.4.6 Applications to System Data

In the standard models of progressive censoring, it is assumed that the underlying random variables are iid random variables distributed according to a cdf F . Progressive censoring has also been studied in terms of iid system data $Y, \dots, Y_n \stackrel{\text{iid}}{\sim} G$, where $G = h \circ F$ with some known function h and a cdf F . F is supposed to model the component lifetime, whereas h describes the technical structure of the system. In a series system with s components, we have $G = F^s$ with $h(t) = t^s$. For a parallel system with s components, the function h is given by $h(t) = 1 - (1 - t)^s$.

As pointed out in [1, Chapter 25], the case of progressively type II censored series system data is included in the standard model by adapting the censoring plan appropriately. Given a censoring plan $\mathcal{R} = (R_1, \dots, R_m)$ and series systems with s components, the corresponding progressively type II censored system data can be interpreted as standard progressively type II censored data with censoring plan

$$\mathcal{S} = s\mathcal{R} + (s - 1)(1^{*m}), \text{ i.e., } S_j = sR_j + s - 1, j = 1, \dots, m.$$

This kind of data has also been entitled *first-failure censored data* (see, e.g., [326, 327]), and many results have been published for this model. However, as mentioned before, the respective results are covered in the standard model by adapting the censoring plan.

The situation is more involved for parallel or, more general, for coherent systems. Parallel systems are studied in [166,328,329], and [330]. k out-of- n system data is addressed in [331], and coherent systems are addressed in [332].

9.4.7 Applications in Quality Control

Applications of progressively censored data in quality control have been discussed in terms of reliability sampling plans (acceptance sampling plans) and the lifetime performance index, respectively.

Reliability sampling plans based on progressively type II censored exponential lifetimes have been considered by Balasooriya and Saw [333], Balakrishnan and Aggarwala [2], and [334] [see also 1, Chapter 22]. For a progressively censored sample $X^{\mathcal{R}}$ from an $\text{Exp}(\mu, \vartheta)$ -distribution, the MLEs of the parameters are used to estimate the parameters and, thus, to construct the decision rule. Since the distributions of the MLEs do not depend on the censoring plan \mathcal{R} , the resulting sampling plans coincide with those for type II right censoring. Pérez-González and Fernández [335] established approximate acceptance sampling plans in the two-parameter exponential case. Balasooriya et al. [147] addressed reliability sampling plans for a Weibull distribution employing the Lieberman-Resnikoff procedure for a lower limit using the approximate MLEs of the parameters. Ng et al. [139] tackled the same problem using the MLEs. Fernández et al. [336] considered progressively censored group sampling plans for Weibull distributions. For a log-normal distribution, [337] applied the Lieberman-Resnikoff approach using approximate BLUEs for the location and scale parameters. Further reference in this direction is [338].

Inference for the lifetime performance index (or capability index) has been discussed for various distributions in [339] including exponential and gamma distributions (see also [340–343]). Weibull distribution has been considered in [344, 345], and Rayleigh distribution in [341]. Lomax and Pareto distributions are discussed by Mahmoud et al. [346] and Ahmadi and Doostparast [347], respectively. Progressively type I interval censored data has been considered in [348–350].

9.4.8 Accelerated Life Testing

In accelerated life testing, progressive censoring has been mostly discussed in terms of step-stress testing. Recent reviews on the topic are provided by Kundu and Ganguly [351] and Balakrishnan and Cramer [1, Chapter 23] (see also [352]). Assuming a cumulative exposure model for the lifetime distribution, the basic model in simple step-stress model with a single stress change point is applied to progressively

type II censored data, that is, at a prefixed time τ , the stress level is to be increased to a level $s_1 > s_0$ (see Fig. 9.6). Then, the data

$$X_{1:r:n} < \cdots < X_{D:r:n} \leq \tau < X_{D+1:r:n} < \cdots < X_{r:r:n}$$

results where D denotes the number of failures observed before τ . Obviously, the sample $X_{1:r:n}, \dots, X_{D:r:n}$ is a type I progressive hybrid censored sample so that the inferential results can be taken from this area. Assuming exponential lifetimes with means ϑ_1 and ϑ_2 (before and after τ) as well as a cumulative exposure model, the MLEs of the parameters are given by

$$\hat{\vartheta}_1 = \frac{1}{D} \left(\sum_{k=1}^D (R_k + 1) X_{k:r:n} + \tau \gamma_{D+1} \right),$$

$$\hat{\vartheta}_2 = \frac{1}{r-D} \sum_{k=D+1}^r (R_k + 1) (X_{k:r:n} - \tau),$$

provided that $1 \leq D \leq r-1$. As mentioned above, distributional results for $\hat{\vartheta}_1$ are directly obtained from type I progressive hybrid censoring leading to, e.g., exact (conditional) confidence intervals (see [353, 354]). Using the result in (9.6), we get $2(r-D)\hat{\vartheta}_2/\vartheta_2|D = d \sim \chi^2(2r-2d)$, $d < r$. In particular, $E(\hat{\vartheta}_2|D = d) = \vartheta_2$ and $\text{Var}(\hat{\vartheta}_2|D = d) = \vartheta_2^2/(r-d)$. Weibull lifetimes are investigated in [355]. Extensions to multiple stress changing times are discussed in [356, 357]. The model has also been discussed for progressive type I interval censored data (see [358–360]).

Wang and Yu [361] discussed a simple step-stress model where the stress changing time τ is replaced by a failure time $X_{r_1:r:n}$. It is shown in [1, p. 492] that this model is connected to sequential order statistics which can be utilized to establish easily properties of the resulting MLEs. Following the ideas of [362], a multiple step-stress model with additional progressive censoring has been proposed in [1, p. 503]. The resulting model corresponds to that proposed in [361] and is further discussed in [363].

Another kind of accelerated life testing model in the presence of progressive censoring, called progressive stress model, has been proposed by Abdel-Hamid and AL-Hussaini [364].

9.4.9 Stage Life Testing

A common argument used to justify progressive censoring is that the intentionally removed objects are utilized for other tests (cf., e.g., [2, p. 3], [365, p. 336]). In order to include such an information in the statistical analysis, [366] connected the notion of progressive censoring with ideas from accelerated

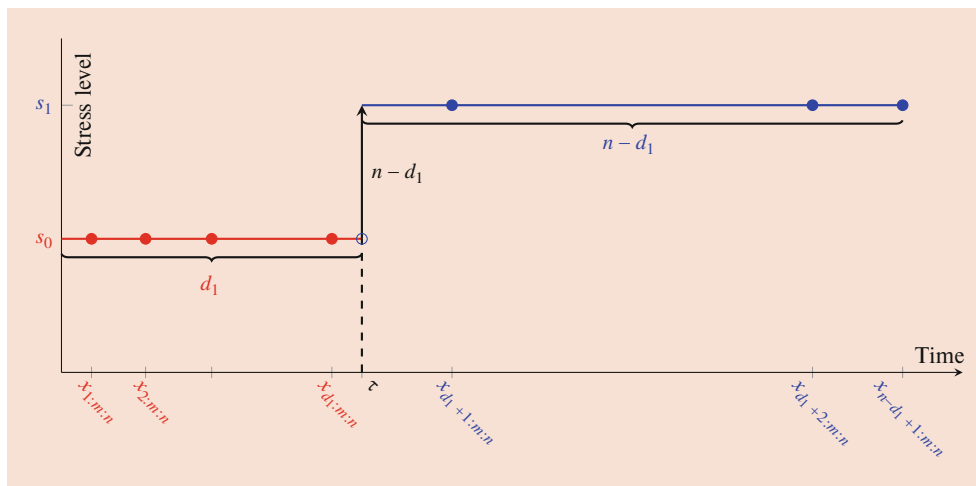


Fig. 9.6 Step-stress testing with single stress change time τ

life testing and proposed a stage life testing model with a single stage change time. In fact, they adapted the idea of step-stress testing and assumed that the *removed* objects are tested on a different level, whereas the remaining items are still tested under standard conditions. The basic situation with a single censoring time τ and two stages s_0, s_1 is depicted in Fig. 9.7.

Inspired by simple step-stress testing (see Fig. 9.6), the stage life testing model is introduced as follows. Assume that n identical objects are placed on a life test. The initial conditions are called stage s_0 . At the prefixed stage-change time τ , $0 \leq R_1^* \leq n$ of the surviving items are randomly withdrawn (if possible) and further tested on stage s_1 . Notice that this may be regarded as a different life test with used components. The testing of the remaining items is continued on stage s_0 . The life test terminates when all n objects have failed.

Let D_1 and D_2 denote the random number of failures occurring on stage s_0 before and after τ_1 , respectively. Furthermore, $M = D_1 + D_2$ and

- $\mathbf{Y}_{1,D_1} = (Y_{1:M:n}, \dots, Y_{D_1:M:n})$ denote the (ordered) observations on stage s_0 before τ .
- $\mathbf{Y}_{2,D_2} = (Y_{D_1+1:M:n}, \dots, Y_{D_1+D_2:M:n})$ denote the (ordered) observations on stage s_0 after τ .
- $\mathbf{Z}_{R_1^*} = (Z_{1:R_1^*}, \dots, Z_{R_1^*:R_1^*})$ denote the (ordered) observations on stage s_1 after τ with $Y_{D_1:M:n} \leq \tau < Z_{1:R_1^*}$.

The order statistics on stage s_0 and the order statistics on stage s_1 are represented by the random vectors $\mathbf{Y} = (\mathbf{Y}_{1,D_1}, \mathbf{Y}_{2,D_2})$ and \mathbf{Z} , respectively. Figure 9.7 illustrates this representation of the *stage life testing* order statistics.

Assuming a cumulative exposure model with cdfs F_0 and F_1 and utilizing progressive censoring with fixed censoring times (see [6]), inferential results have been obtained for

exponential, Weibull, and other lifetime distributions. The results are based on the joint pdf $f_{1..n}^{Y,Z,D_1}$ of (\mathbf{Y}, \mathbf{Z}) and D_1 given by

$$\begin{aligned} f_{1..n}^{Y,Z,D_1}(\mathbf{y}_{1,d_1}, \mathbf{y}_{2,d_2}, \mathbf{z}, d_1) \\ = \binom{n}{d_1} d_1! d_2! r_1^*! \prod_{h=1}^{d_1+d_2} f_0(y_{h:m:n}) \prod_{j=1}^{r_1^*} f_1(z_{j:r_1^*} + v_1 - \tau), \end{aligned}$$

for $\mathbf{y}_{1,d_1} = (y_{1:m:n}, \dots, y_{d_1:m:n})$, $\mathbf{y}_{2,d_2} = (y_{d_1+1:m:n}, \dots, y_{d_1+d_2:m:n})$, and $\mathbf{z} = (z_{1:r_1^*}, \dots, z_{r_1^*:r_1^*})$, where $d_2 = n - d_1 - r_1^*$, and $r_1^* = \varrho(d_1)$ denotes the number of objects selected for testing on stage s_1 . f_i denotes the pdf of F_i , $i \in \{0, 1\}$.

In fact, the function ϱ may be chosen according to the needs of the experimenter. Inspired by procedures to generate the censoring number in progressive censoring (see [7]), [366] proposed two options to define R_1^* , that is,

$$R_1^* = \varrho(D_1) \quad \text{with}$$

$$\varrho(x) = \begin{cases} \lfloor \pi_1 \cdot (n - x) \rfloor, & \text{Type-P} \\ \min \{n - x, R_1^0\}, & \text{Type-M} \end{cases}, \quad x \in \{0, \dots, n\},$$

where the proportion $\pi_1 \in [0, 1]$ and the number $R_1^0 \in \mathbb{N}$ are prespecified, respectively. Under the type P scheme, at τ , a (fixed) proportion π_1 of the surviving objects is selected for testing on stage s_1 . In case of the type M scheme, given a prefixed number R_1^0 , it is intended to select at τ as many items as possible (at most R_1^0) for testing on stage s_1 .

Assuming exponential lifetimes with means ϑ_0 and ϑ_1 , the corresponding MLEs are obtained as

$$\hat{\vartheta}_0 = \frac{1}{D_1 + D_2} \left(\sum_{i=1}^{D_1+D_2} Y_{i:M:n} + R_1^* \tau \right),$$

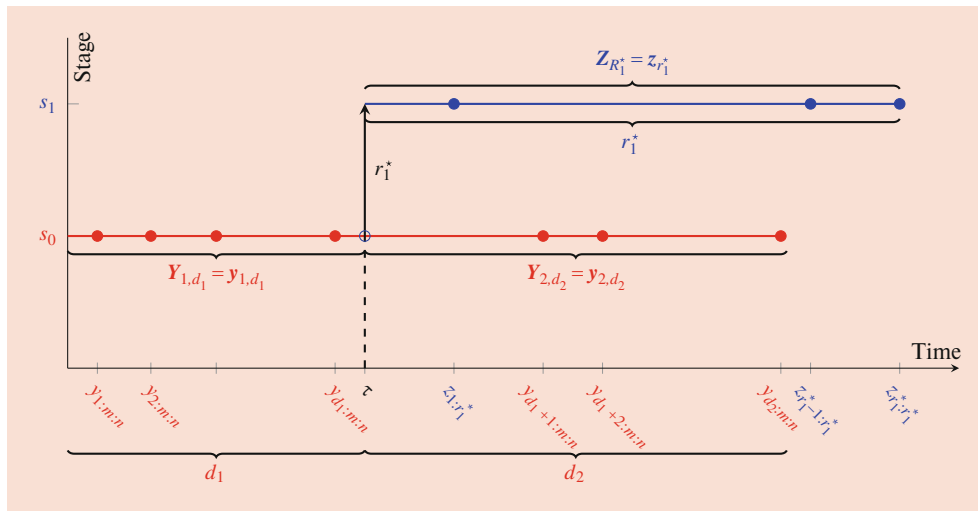


Fig. 9.7 Stage life testing with single stage change time τ

$$\hat{\vartheta}_1 = \frac{1}{R_1^*} \left(\sum_{j=1}^{R_1^*} Z_{j:R_1^*} - R_1^* \tau \right) \text{ (provided that } R_1^* > 0 \text{)}.$$

As for other progressive censoring models, exact inferential results, e.g., exact confidence intervals, can be established. An extension to multiple stage change times $\tau_1 < \dots < \tau_m$ has been introduced in [367]. Stage changes at failure times are discussed in [368]. For a survey on these models as well as more details, we refer to [369].

9.4.10 Joint Progressive Censoring Schemes

In joint progressive type II censoring, the sample is based on two baseline samples $X_1, \dots, X_{n_1} \stackrel{iid}{\sim} F_1$ (product/type A) and $Y_1, \dots, Y_{n_2} \stackrel{iid}{\sim} F_2$ (product/type B) of independent random variables. The progressive censoring is applied to the pooled sample $X_1, \dots, X_{n_1}, Y_1, \dots, Y_{n_2}$ given a prefixed censoring $\mathcal{R} \in \mathcal{C}_{m, n_1+n_2}^m$. Moreover, it is assumed that the type of the failed unit as well as the types of withdrawn units are known. Therefore, the sample is given by $(\mathbf{C}, \mathbf{W}^{\mathcal{R}}, \mathcal{S})$, where

$$\mathbf{C} = (C_1, \dots, C_m) \in \{0, 1\}^m,$$

$$\mathbf{W}^{\mathcal{R}} = (W_{1:m:n_1+n_2}, \dots, W_{m:m:n_1+n_2}), \mathcal{S} = (S_1, \dots, S_m).$$

The indicators C_j have the value 1 if the failed unit is of type A, and otherwise $C_j = 0$. $W_{j:m:n}$ denotes the j th failure time in the progressively censored experiment. Finally, \mathcal{S} denotes a random censoring plan. S_j is the number of removed units of type A in the j th withdrawal. Thus, $R_j - S_j$ denotes the numbers of withdrawn units of type B at the j th censoring step. This model has been discussed in [370–376]. Joint

progressive type I censoring has been proposed in [377]. A hybrid version is discussed in [378]. Another version of joint progressive censoring leading to more tractable results has been investigated in [376] and [379].

References

1. Balakrishnan, N., Cramer, E.: The Art of Progressive Censoring. Applications to Reliability and Quality. Birkhäuser, New York (2014)
2. Balakrishnan, N., Aggarwala, R.: Progressive Censoring: Theory, Methods, and Applications. Birkhäuser, Boston (2000)
3. Balakrishnan, N.: Progressive censoring methodology: an appraisal (with Discussions). TEST **16**, 211–296 (2007)
4. Cramer, E.: Progressive Censoring Schemes. In: Wiley StatsRef: Statistics Reference Online. Wiley, New York (2017)
5. Cohen, A.C.: Progressively censored samples in life testing. Technometrics **5**, 327–329 (1963)
6. Laumen, B., Cramer, E.: Progressive censoring with fixed censoring times. Statistics **53**, 569–600 (2019)
7. Aggarwala, R.: Progressive interval censoring: some mathematical results with applications to inference. Commun. Statist. Theory Methods **30**(8-9), 1921–1935 (2001)
8. Childs, A., Chandrasekar, B., Balakrishnan, N.: Exact likelihood inference for an exponential parameter under progressive hybrid censoring schemes. In: Vonta, F., Nikulin, M., Limnios, N., Huber-Carol, C. (eds.) Statistical Models and Methods for Biomedical and Technical Systems, pp. 323–334. Birkhäuser, Boston (2008)
9. Kundu, D., Joarder, A.: Analysis of Type-II progressively hybrid censored data. Comput. Statist. Data Anal. **50**(10), 2509–2528 (2006)
10. Cramer, E., Balakrishnan, N.: On some exact distributional results based on Type-I progressively hybrid censored data from exponential distributions. Statist. Methodol. **10**, 128–150 (2013)
11. Epstein, B.: Truncated life tests in the exponential case. Ann. Math. Stat. **25**, 555–564 (1954)
12. Górný, J., Cramer, E.: Modularization of hybrid censoring schemes and its application to unified progressive hybrid censoring. Metrika **81**, 173–210 (2018)

13. Ng, H.K.T., Kundu, D., Chan, P.S.: Statistical analysis of exponential lifetimes under an adaptive Type-II progressive censoring scheme. *Naval Res. Logist.* **56**, 687–698 (2009)
14. Cramer, E., Iliopoulos, G.: Adaptive progressive Type-II censoring. *TEST* **19**, 342–358 (2010)
15. Cramer, E., Iliopoulos, G.: Adaptive Progressive Censoring. In: Choudhary, P.K., Nagaraja, C.H., Ng, H.K.T. (eds.) *Ordered Data Analysis, Modeling and Health Research Methods—In Honor of H.N. Nagaraja's 60th Birthday*, pp. 73–86. Springer, New York (2015)
16. Balakrishnan, N., Cramer, E.: Progressive censoring from heterogeneous distributions with applications to robustness. *Ann. Inst. Statist. Math.* **60**, 151–171 (2008)
17. Fischer, T., Balakrishnan, N., Cramer, E.: Mixture representation for order statistics from INID progressive censoring and its applications. *J. Multivariate Anal.* **99**, 1999–2015 (2008)
18. Cramer, E., Lenz, U.: Association of progressively Type-II censored order statistics. *J. Statist. Plann. Inference* **140**(2), 576–583 (2010)
19. Rezapour, M., Alamatsaz, M.H., Balakrishnan, N., Cramer, E.: On properties of progressively Type-II censored order statistics arising from dependent and non-identical random variables. *Statist. Methodol.* **10**(1), 58–71 (2013)
20. Rezapour, M., Alamatsaz, M.H., Balakrishnan, N.: On properties of dependent progressively Type-II censored order statistics. *Metrika* **76**, 909–917 (2013)
21. Kamps, U.: *A Concept of Generalized Order Statistics*. Teubner, Stuttgart (1995)
22. Kamps, U.: A concept of generalized order statistics. *J. Statist. Plann. Inference* **48**, 1–23 (1995)
23. Cramer, E., Kamps, U.: Sequential k -out-of- n systems. In: Balakrishnan, N., Rao, C.R. (eds.) *Handbook of Statistics: Advances in Reliability*, vol. 20, pp. 301–372. Elsevier, Amsterdam (2001). Chap. 12
24. Thomas, D.R., Wilson, W.M.: Linear order statistic estimation for the two-parameter Weibull and extreme value distributions from Type-II progressively censored samples. *Technometrics* **14**, 679–691 (1972)
25. Viveros, R., Balakrishnan, N.: Interval estimation of parameters of life from progressively censored data. *Technometrics* **36**, 84–91 (1994)
26. Kamps, U., Cramer, E.: On distributions of generalized order statistics. *Statistics* **35**, 269–280 (2001)
27. Balakrishnan, N., Cramer, E., Kamps, U.: Relation for joint densities of progressively censored order statistics. *Statistics* **39**(6), 529–536 (2005)
28. Cramer, E.: Dependence structure of generalized order statistics. *Statistics* **40**, 409–413 (2006)
29. Balakrishnan, N., Dembińska, A.: Progressively Type-II right censored order statistics from discrete distributions. *J. Statist. Plann. Inference* **138**, 845–856 (2008)
30. Balakrishnan, N., Dembińska, A.: Erratum to 'Progressively Type-II right censored order statistics from discrete distributions' [*J. Statist. Plann. Inference* 138 (2008) 845–856]. *J. Statist. Plann. Inference* **139**, 1572–1574 (2009)
31. Schenk, N.: *Point Estimation with Sequential Order Statistics from Exponential Distributions* Ph.D. Thesis. University of Oldenburg, Oldenburg (2001)
32. Cramer, E., Kamps, U., Rychlik, T.: Unimodality of uniform generalized order statistics, with applications to mean bounds. *Ann. Inst. Statist. Math.* **56**, 183–192 (2004)
33. Cramer, E.: Logconcavity and unimodality of progressively censored order statistics. *Statist. Probab. Letters* **68**, 83–90 (2004)
34. Chen, H., Xie, H., Hu, T.: Log-concavity of generalized order statistics. *Statist. Probab. Letters* **79**, 396–399 (2009)
35. Alimohammadi, M., Alamatsaz, M.H.: Some new results on unimodality of generalized order statistics and their spacings. *Statist. Probab. Lett.* **81**(11), 1677–1682 (2011)
36. Alimohammadi, M., Alamatsaz, M.H., Cramer, E.: Convolutions and generalization of logconcavity: Implications and applications. *Nav. Res. Logist. (NRL)* **63**(2), 109–123 (2016)
37. Cramer, E., Kamps, U.: Marginal distributions of sequential and generalized order statistics. *Metrika* **58**, 293–310 (2003)
38. Balakrishnan, N., Sandhu, R.A.: A simple simulational algorithm for generating progressive Type-II censored samples. *Am. Statist.* **49**, 229–230 (1995)
39. Iliopoulos, G., Balakrishnan, N.: Conditional independence of blocked ordered data. *Statist. Probab. Letters* **79**, 1008–1015 (2009)
40. Arnold, B.C., Balakrishnan, N., Nagaraja, H.N.: *A First Course in Order Statistics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2008)
41. David, H.A., Nagaraja, H.N.: *Order Statistics*, 3rd edn. Wiley, Hoboken (2003)
42. Cramer, E., Kamps, U., Rychlik, T.: On the existence of moments of generalized order statistics. *Statist. Probab. Letters* **59**, 397–404 (2002)
43. Cramer, E.: A note on moments of progressively Type II censored order statistics. *Commun. Statist. Theory Methods* **31**, 1301–1307 (2002)
44. Balakrishnan, N., Cramer, E., Kamps, U.: Bounds for means and variances of progressive type II censored order statistics. *Statist. Probab. Letters* **54**, 301–315 (2001)
45. Raqab, M.Z.: P-norm bounds for moments of progressive type II censored order statistics. *Statist. Probab. Letters* **64**(4), 393–402 (2003)
46. Aggarwala, R., Balakrishnan, N.: Recurrence relations for single and product moments of progressive type-II right censored order statistics from exponential and truncated exponential distributions. *Ann. Inst. Statist. Math.* **48**, 757–771 (1996)
47. Balasooriya, U., Saw, S.L.C.: A note on approximate moments of progressively censored order statistics. *Metron* **57**(1-2), 117–130 (1999)
48. Pérez-González, C.J., Fernández, A.J.: Accurate computation of single and product moments of order statistics under progressive censoring. *J. Stat. Comput. Simul.* **89**(13), 2489–2504 (2019)
49. Cramer, E., Davies, K.: Restricted optimal progressive censoring. *Commun. Stat. Simul. Comput.* **47**, 1216–1239 (2018)
50. Shaked, M., Shanthikumar, J.G.: *Stochastic Orders*. Springer, Berlin (2007)
51. Müller, A., Stoyan, D.: *Comparison Methods for Stochastic Models and Risks*. Wiley, Chichester (2002)
52. Belzunce, F., Riquelme, C., Mulero, J.: *An Introduction to Stochastic Orders*. Academic Press, London (2015)
53. Belzunce, F.: Multivariate comparisons of ordered data. In: *Stochastic Orders in Reliability and Risk: In: Li, H., Li, X. (eds.) Honor of Professor Moshe Shaked*, pp. 83–102. Springer, New York (2013)
54. Belzunce, F., Mercader, J.-A., Ruiz, J.-M.: Stochastic comparisons of generalized order statistics. *Probab. Engrg. Inform. Sci.* **19**, 99–120 (2005)
55. Khaledi, B.-E.: Some new results on stochastic orderings between generalized order statistics. *J. Iran. Stat. Soc. (JIRSS)* **4**, 35–49 (2005)
56. Bon, J.-L., Păltănea, E.: Ordering properties of convolutions of exponential random variables. *Lifetime Data Anal.* **5**(2), 185–192 (1999)
57. Khaledi, B.-E., Kochar, S.C.: Stochastic ordering among order statistics and sample spacings. In: *Misra, J.C. (ed.) Uncertainty*

- and Optimality—Probability, Statistics and Operations Research, pp. 167–203. World Scientific Publications, Singapore (2002)
58. Burkschat, M.: On optimality of extremal schemes in progressive Type-II censoring. *J. Statist. Plann. Inference* **138**, 1647–1659 (2008)
 59. Hu, T., Zhuang, W.: A note on stochastic comparisons of generalized order statistics. *Statist. Probab. Letters* **72**(2), 163–170 (2005)
 60. Korwar, R.: On the likelihood ratio order for progressive type II censored order statistics. *Sankhyā A* **65**, 793–798 (2003)
 61. Cramer, E., Kamps, U., Raqab, M.Z.: Characterizations of exponential distributions by spacings of generalized order statistics. *Appl. Math.* **30**, 257–265 (2003)
 62. Zhuang, W., Hu, T.: Multivariate dispersive ordering of spacings of generalized order statistics. *Appl. Math. Letters* **22**(6), 968–974 (2009)
 63. Balakrishnan, N., Belzunce, F., Hami, N., Khaledi, B.-E.: Univariate and multivariate likelihood ratio ordering of generalized order statistics and associated conditional variables. *Probab. Eng. Inform. Sci.* **24**, 441–455 (2010)
 64. Sharafi, M., Khaledi, B.-E., Hami, N.: On multivariate likelihood ratio ordering among generalized order statistics and their spacings. *J. Iranian Stat. Soc.* **13**, 1–29 (2014)
 65. Arriaza, A., Belzunce, F., Mulero, J., Suárez-Llorens, A.: On a new multivariate IFR ageing notion based on the standard construction. *Appl. Stoch. Model. Bus. Ind.* **32**(2), 292–306 (2016)
 66. Belzunce, F., Lillo, R.E., Ruiz, J.-M., Shaked, M.: Stochastic comparisons of nonhomogeneous processes. *Probab. Eng. Inform. Sci.* **15**(2), 199–224 (2001)
 67. Xie, H., Hu, T.: Conditional ordering of generalized order statistics revisited. *Probab. Eng. Inform. Sci.* **22**, 333–346 (2008)
 68. Xie, H., Hu, T.: Ordering p -spacings of generalized order statistics revisited. *Probab. Eng. Inform. Sci.* **23**, 1–16 (2009)
 69. Belzunce, F., Ruiz, J.M., Suárez-Llorens, A.: On multivariate dispersion orderings based on the standard construction. *Statist. Probab. Letters* **78**(3), 271–281 (2008)
 70. Chen, J., Hu, T.: Multivariate dispersive ordering of generalized order statistics. *J. Iran. Stat. Soc. (JIRSS)* **6**, 61–75 (2007)
 71. Xie, H., Hu, T.: Some new results on multivariate dispersive ordering of generalized order statistics. *J. Multivariate Anal.* **101**(4), 964–970 (2010)
 72. Hashemi, M., Tavangar, M., Asadi, M.: Some properties of the residual lifetime of progressively Type-II right censored order statistics. *Statist. Probab. Letters* **80**(9–10), 848–859 (2010)
 73. Xie, H., Zhuang, W.: Some new results on ordering of simple spacings of generalized order statistics. *Probab. Eng. Inform. Sci.* **25**(01), 71–81 (2011)
 74. Belzunce, F., Martínez-Riquelme, C.: Some results for the comparison of generalized order statistics in the total time on test and excess wealth orders. *Stat. Pap.* **56**(4), 1175–1190 (2015)
 75. Belzunce, F., Martínez-Riquelme, C., Ruiz, J.M., Sordo, M.A.: On sufficient conditions for the comparison in the excess wealth order and spacings. *J. Appl. Probab.* **53**(1), 33–46 (2016)
 76. Balakrishnan, N., Belzunce, F., Sordo, M.A., Suárez-Llorens, A.: Increasing directionally convex orderings of random vectors having the same copula, and their use in comparing ordered data. *J. Multivariate Anal.* **105**(1), 45–54 (2012)
 77. Zhao, P., Balakrishnan, N.: Stochastic comparisons and properties of conditional generalized order statistics. *J. Statist. Plann. Inference* **139**(9), 2920–2932 (2009)
 78. Tavangar, M., Bairamov, I.: A note on the residual lifetimes in a life-test under progressive type-II right censoring scheme. *Appl. Comput. Math.* **11**, 368–377 (2012)
 79. Mao, T., Hu, T.: Stochastic properties of INID progressive Type-II censored order statistics. *J. Multivariate Anal.* **101**(6), 1493–1500 (2010)
 80. Barlow, R.E., Proschan, F.: *Mathematical Theory of Reliability*. Wiley, New York (1965)
 81. Lai, C.-D., Xie, M.: *Stochastic Ageing and Dependence for Reliability*. Springer, New York (2006)
 82. Marshall, A.W., Olkin, I.: *Life Distributions. Structure of Non-parametric, Semiparametric, and Parametric Families*. Springer, New York (2007)
 83. Burkschat, M., Navarro, J.: Aging properties of sequential order statistics. *Probab. Eng. Inform. Sci.* **25**(04), 449–467 (2011)
 84. Torrado, N., Lillo, R., Wiper, M.: Sequential order statistics: ageing and stochastic orderings. *Method. Comput. Appl. Probab.* **14**, 579–596 (2012)
 85. Burkschat, M., Torrado, N.: On the reversed hazard rate of sequential order statistics. *Statist. Probab. Lett.* **85**, 106–113 (2014)
 86. Belzunce, F., Mercader, J.A., Ruiz, J.M.: Multivariate aging properties of epoch times of nonhomogeneous processes. *J. Multivariate Anal.* **84**(2), 335–350 (2003)
 87. Arias-Nicolás, J.P., Belzunce, F., Núñez Barrera, O., Suárez-Llorens, A.: A multivariate IFR notion based on the multivariate dispersive ordering. *Appl. Stoch. Models Bus. Ind.* **25**(3), 339–358 (2009)
 88. Tavangar, M., Asadi, M.: On stochastic and aging properties of generalized order statistics. *Probab. Eng. Inform. Sci.* **25**(02), 187–204 (2011)
 89. Lin, C.-T., Balakrishnan, N.: Asymptotic properties of maximum likelihood estimators based on progressive Type-II censoring. *Metrika* **74**, 349–360 (2011)
 90. Hoadley, B.: Asymptotic properties of maximum likelihood estimators for the independent not identically distributed case. *Ann. Math. Stat.* **42**(6), 1977–1991 (1971)
 91. Cramer, E.: Extreme value analysis for progressively Type-II censored order statistics. *Commun. Statist. Theory Methods* **43**, 2135–2155 (2014)
 92. Cramer, E.: *Contributions to Generalized Order Statistics. Habilitationsschrift*. University of Oldenburg, Oldenburg (2003)
 93. Schmiedt, A.B.: Domains of attraction of asymptotic distributions of extreme generalized order statistics. *Commun. Stat. Theory and Methods* **45**(7), 2089–2104 (2016)
 94. Bordes, L.: Non-parametric estimation under progressive censoring. *J. Statist. Plann. Inference* **119**(1), 171–189 (2004)
 95. Alvarez-Andrade, S., Bordes, L.: Empirical quantile process under type-II progressive censoring. *Statist. Probab. Letters* **68**(1), 111–123 (2004)
 96. Alvarez-Andrade, S., Bordes, L.: Type-II progressive censoring and related processes. *Rev. Roumaine Math. Pures Appl.* **63**, 267–276 (2008)
 97. Hofmann, G., Cramer, E., Balakrishnan, N., Kunert, G.: An asymptotic approach to progressive censoring. *J. Statist. Plann. Inference* **130**(1-2), 207–227 (2005)
 98. Ng, H.K.T., Chan, P.S., Balakrishnan, N.: Estimation of parameters from progressively censored data using EM algorithm. *Comput. Statist. Data Anal.* **39**(4), 371–386 (2002)
 99. Zheng, G., Park, S.: On the Fisher information in multiply censored and progressively censored data. *Commun. Statist. Theory Methods* **33**(8), 1821–1835 (2004)
 100. Balakrishnan, N., Burkschat, M., Cramer, E., Hofmann, G.: Fisher information based progressive censoring plans. *Comput. Statist. Data Anal.* **53**, 366–380 (2008)
 101. Abo-Eleneen, Z.A.: Fisher information in type II progressive censored samples. *Commun. Statist. Theory Methods* **37**, 682–691 (2008)
 102. Park, S., Ng, H.K.T.: Missing information and an optimal one-step plan in a Type II progressive censoring scheme. *Statist. Probab. Letters* **82**(2), 396–402 (2012)
 103. Burkschat, M., Cramer, E.: Fisher information in generalized order statistics. *Statistics* **46**, 719–743 (2012)

104. Dahmen, K., Burkschat, M., Cramer, E.: A- and D-optimal progressive Type-II censoring designs based on Fisher information. *J. Stat. Comput. Simul.* **82**, 879–905 (2012)
105. Cramer, E., Ensenbach, M.: Asymptotically optimal progressive censoring plans based on Fisher information. *J. Statist. Plann. Inference* **141**, 1968–1980 (2011)
106. Balakrishnan, N., Habibi Rad, A., Arghami, N.R.: Testing exponentiality based on Kullback-Leibler information with progressively Type-II censored data. *IEEE Trans. Reliab.* **56**, 301–307 (2007)
107. Cramer, E., Bagh, C.: Minimum and maximum information censoring plans in progressive censoring. *Commun. Statist. Theory Methods* **40**, 2511–2527 (2011)
108. Ahmadi, J.: Some results based on entropy properties of progressive Type-II censored data. *J. Statist. Res. Iran* **4**, 191–202 (2007)
109. Abo-Eleneen, Z.A.: The entropy of progressively censored samples. *Entropy* **13**(2), 437–449 (2011)
110. Abo-Eleneen, Z., Almohaimeed, B., Ng, H.K.T.: On cumulative residual entropy of progressively censored order statistics. *Statist. Probab. Lett.* **139**, 47–52 (2018)
111. Habibi Rad, A., Yousefzadeh, F., Balakrishnan, N.: Goodness-of-fit test based on Kullback-Leibler Information for progressively Type-II censored data. *IEEE Trans. Reliab.* **60**, 570–579 (2011)
112. Baratpour, S., Habibi Rad, A.: Exponentiality test based on the progressive type II censoring via cumulative entropy. *Commun. Stat. Simul. Comput.* **45**(7), 2625–2637 (2016)
113. Park, S., Pakyari, R.: Cumulative residual Kullback-Leibler information with the progressively Type-II censored data. *Statist. Probab. Lett.* **106**, 287–294 (2015)
114. Döring, M., Cramer, E.: On the power of goodness-of-fit tests for the exponential distribution under progressive Type-II censoring. *J. Stat. Comput. Simul.* **89**, 2997–3034 (2019)
115. Zhang, Y., Gui, W.: A goodness of fit test for the Pareto distribution with progressively Type II censored data based on the cumulative hazard function. *J. Comput. Appl. Math.* **368**, 112557 (2020)
116. Volterman, W., Davies, K.F., Balakrishnan, N.: Simultaneous Pitman closeness of progressively Type-II right-censored order statistics to population quantiles. *Statistics* **47**, 439–452 (2013)
117. Volterman, W., Davies, K.F., Balakrishnan, N.: Pitman closeness as a criterion for the determination of the optimal progressive censoring scheme. *Statist. Methodol.* **9**(6), 563–572 (2012)
118. Volterman, W., Davies, K.F., Balakrishnan, N.: Two-sample Pitman closeness comparison under progressive Type-II censoring. *Statistics* **47**, 1305–1320 (2013)
119. Raqab, M.Z., Alkhalaf, L.A., Balakrishnan, N.: Pitman comparisons of predictors of censored observations from progressively censored samples for exponential distribution. *J. Stat. Comput. Simul.* **86**(8), 1539–1558 (2016)
120. Bairamov, I., Eryilmaz, S.: Spacings, exceedances and concomitants in progressive Type II censoring scheme. *J. Statist. Plann. Inference* **136**(3), 527–536 (2006)
121. Izadi, M., Khaledi, B.-E.: Progressive Type II censored order statistics and their concomitants: some stochastic comparisons results. *J. Iran. Stat. Soc. (JIRSS)* **6**, 111–124 (2007)
122. Berred, A., Stepanov, A.: Asymptotic properties of the number of near minimum-concomitant observations in the case of progressive type-II censoring. *Metrika* **78**(3), 283–294 (2015)
123. Razmkhah, M., Simriz, S.: Statistical inferences based on INID progressively type II censored order statistics. *Ann. Inst. Stat. Math.* **70**, 583–604 (2018)
124. Bairamov, I.: Progressive type II censored order statistics for multivariate observations. *J. Multivariate Anal.* **97**(4), 797–809 (2006)
125. Amirzadeh, V., Rezapour, M.: On properties of progressively Type-II censored conditionally N-ordered statistics arising from a non-identical and dependent random vector. *J. Stat. Comput. Simul.* **86**, 1818–1828 (2016)
126. Rezapour, M.: Progressively Type-II censored conditionally N-ordered statistics from a unified elliptically contoured copula. *Commun. Stat. Theory Methods* **46**(11), 5595–5611 (2017)
127. Christensen, R.: *Plane Answers to Complex Questions: The Theory of Linear Models*, 4th edn. Springer, New York (2011)
128. Balakrishnan, N., Sandhu, R.A.: Linear estimation under censoring and inference. In: Balakrishnan, N., Basu, A.P. (eds.) *The Exponential Distribution*, pp. 53–72. Gordon and Breach, Amsterdam (1995)
129. Cramer, E., Kamps, U.: Estimation with sequential order statistics from exponential distributions. *Ann. Inst. Statist. Math.* **53**, 307–324 (2001)
130. Balakrishnan, N., Cramer, E., Kamps, U., Schenk, N.: Progressive type II censored order statistics from exponential distributions. *Statistics* **35**, 537–556 (2001)
131. Mann, N.R.: Best linear invariant estimation for Weibull parameters under progressive censoring. *Technometrics* **13**, 521–533 (1971)
132. Cramer, E., Kamps, U.: Sequential k -out-of- n systems with Weibull components. *Econom. Quality Control* **13**, 227–239 (1998)
133. Guillaud, O.: Exact non-parametric confidence intervals for quantiles with progressive Type-II censoring. *Scand. J. Stat.* **28**, 699–713 (2001)
134. Guillaud, O.: Exact non-parametric confidence, prediction and tolerance intervals with progressive type-II censoring. *Scand. J. Stat.* **31**(2), 265–281 (2004)
135. Balakrishnan, N., Burkschat, M., Cramer, E.: Best linear equivariant estimation and prediction in location-scale families. *Sankhyā B* **70**, 229–247 (2008)
136. Burkschat, M.: Linear estimators and predictors based on generalized order statistics from generalized Pareto distributions. *Commun. Statist. Theory Methods* **39**(2), 311–326 (2010)
137. Mann, N.R.: Optimum estimators for linear functions of location and scale parameters. *Ann. Math. Stat.* **40**, 2149–2155 (1969)
138. Balakrishnan, N., Kateri, M.: On the maximum likelihood estimation of parameters of Weibull distribution based on complete and censored data. *Statist. Probab. Letters* **78**, 2971–2975 (2008)
139. Ng, H.K.T., Chan, P.S., Balakrishnan, N.: Optimal progressive censoring plans for the Weibull distribution. *Technometrics* **46**(4), 470–481 (2004)
140. Dey, T., Dey, S., Kundu, D.: On progressively type-II censored two-parameter Rayleigh distribution. *Commun. Stat. Simul. Comput.* **45**(2), 438–455 (2016)
141. Kotb, M., Raqab, M.: Statistical inference for modified Weibull distribution based on progressively Type-II censored data. *Math. Comput. Simul.* **162**, 233–248 (2019)
142. Krishna, H., Kumar, K.: Reliability estimation in Lindley distribution with progressively Type II right censored sample. *Math. Comput. Simulation* **82**(2), 281–294 (2011)
143. Valiollahi, R., Raqab, M.Z., Asgharzadeh, A., Alqallaf, F.A.: Estimation and prediction for power Lindley distribution under progressively Type II right censored samples. *Math. Comput. Simul.* **149**, 32–47 (2018)
144. Ahmed, E.A.: Estimation of some lifetime parameters of generalized Gompertz distribution under progressively type-II censored data. *Appl. Math. Model.* **39**(18), 5567–5578 (2015)
145. Balakrishnan, N., Varadan, J.: Approximate MLEs for the location and scale parameters of the extreme value distribution with censoring. *IEEE Trans. Reliab.* **40**(2), 146–151 (1991)
146. Balakrishnan, N., Kannan, N., Lin, C.T., Wu, S.J.S.: Inference for the extreme value distribution under progressive Type-II censoring. *J. Stat. Comput. Simul.* **74**(1), 25–45 (2004)

147. Balasooriya, U., Saw, S.L.C., Gadag, V.: Progressively censored reliability sampling plans for the Weibull distribution. *Technometrics* **42**, 160–167 (2000)
148. Schenk, N., Burkschat, M., Cramer, E., Kamps, U.: Bayesian estimation and prediction with multiply Type-II censored samples of sequential order statistics from one- and two-parameter exponential distributions. *J. Statist. Plann. Inference* **141**, 1575–1587 (2011)
149. Kundu, D.: Bayesian inference and life testing plan for the Weibull distribution in presence of progressive censoring. *Technometrics* **50**(2), 144–154 (2008)
150. Li, F., Shi, Y., Tian, Y.: Bayesian estimation of a Weibull model based on progressively censored data. *Chinese J. Engrg. Math.* **25**(4), 641–650 (2008)
151. Balakrishnan, N., Bordes, L.: Non-parametric hazard rate estimation under progressive Type-II censoring. In: Balakrishnan, N., Rao, C.R. (eds.) *Advances in Survival Analysis. Handbook of Statistics*, vol. 23, pp. 227–249. Elsevier, Amsterdam (2004)
152. Wu, S.-F.: Interval estimation for the two-parameter exponential distribution under progressive censoring. *Qual. Quant.* **44**, 181–189 (2010)
153. Wu, S.-J.: Estimations of the parameters of the Weibull distribution with progressively censored data. *J. Jap. Stat. Soc.* **32**(2), 155–163 (2002)
154. Wang, B.X., Yu, K., Jones, M.: Inference under progressively Type II right censored sampling for certain lifetime distributions. *Technometrics* **52**, 453–460 (2010)
155. Kuş, C., Kaya, M.F.: Estimation for the parameters of the Pareto distribution under progressive censoring. *Commun. Statist. Theory Methods* **36**(5–8), 1359–1365 (2007)
156. Parsi, S., Ganjali, M., Farsipour, N.S.: Simultaneous confidence intervals for the parameters of Pareto distribution under progressive censoring. *Commun. Statist. Theory Methods* **39**, 94–106 (2009)
157. Wu, S.-F.: Interval estimation for the Pareto distribution based on the progressive Type II censored sample. *J. Stat. Comput. Simul.* **80**(4), 463–474 (2010)
158. Fernández, A.J.: Computing optimal confidence sets for Pareto models under progressive censoring. *J. Comput. Appl. Math.* **258**, 168–180 (2014)
159. Asgharzadeh, A., Fernández, A.J., Abdi, M.: Confidence sets for the two-parameter Rayleigh distribution under progressive censoring. *Appl. Math. Model.* **47**, 656–667 (2017)
160. Wang, L.: Optimal interval estimation for a family of lower truncated distribution under progressive censoring. *J. Comput. Appl. Math.* **287**, 67–77 (2015)
161. Bedbur, S., Kamps, U., Lennartz, J.: On a smallest confidence region for a location–scale parameter in progressively type-II censored lifetime experiments. *Statist. Probab. Lett.* **154**, 108545 (2019)
162. Balakrishnan, N., Beutner, E., Cramer, E.: Exact two-sample non-parametric confidence, prediction, and tolerance intervals based on ordinary and progressively Type-II right censored data. *TEST* **19**, 68–91 (2010)
163. Volterman, W., Balakrishnan, N., Cramer, E.: Exact meta-analysis of several independent progressively Type-II censored data. *Appl. Math. Model.* **38**, 949–960 (2014)
164. Weerahandi, S.: *Generalized Inference in Repeated Measures. Exact Methods in MANOVA and Mixed Models*. Wiley, Hoboken (2004)
165. Aggarwala, R., Childs, A.: Conditional inference for the parameters of Pareto distributions when observed samples are progressively censored. In: Balakrishnan, N., Melas, V.B., Ermakov, S. (eds.) *Advances in Stochastic Simulation Methods*, pp. 293–302. Birkhäuser, Boston (2000). Chap. 17
166. Pradhan, B.: Point and interval estimation for the lifetime distribution of a k -unit parallel system based on progressively Type-II censored data. *Econom. Quality Control* **22**, 175–186 (2007)
167. Sultan, K.S., Alsadat, N.H., Kundu, D.: Bayesian and maximum likelihood estimations of the inverse Weibull parameters under progressive type-II censoring. *J. Stat. Comput. Simul.* **84**, 2248–2265 (2014)
168. Basak, I., Basak, P., Balakrishnan, N.: On some predictors of times to failure of censored items in progressively censored samples. *Comput. Statist. Data Anal.* **50**(5), 1313–1337 (2006)
169. Balakrishnan, N., Rao, C.R.: Large-sample approximations to the best linear unbiased estimation and best linear unbiased prediction based on progressively censored samples and some applications. In: Panchapakesan, S., Balakrishnan, N. (eds.) *Advances in Statistical Decision Theory and Applications*, pp. 431–444. Birkhäuser, Boston (1997)
170. Basak, I., Balakrishnan, N.: Predictors of failure times of censored units in progressively censored samples from normal distribution. *Sankhyā* **71-B**, 222–247 (2009)
171. Raqab, M.Z., Asgharzadeh, A., Valiollahi, R.: Prediction for Pareto distribution based on progressively Type-II censored samples. *Comput. Stat. Data Anal.* **54**(7), 1732–1743 (2010)
172. Madi, M.T., Raqab, M.Z.: Bayesian inference for the generalized exponential distribution based on progressively censored data. *Commun. Statist. Theory Methods* **38**(12), 2016–2029 (2009)
173. Raqab, M.Z., Madi, M.T.: Inference for the generalized Rayleigh distribution based on progressively censored data. *J. Statist. Plann. Inference* **141**(10), 3313–3322 (2011)
174. Abdel-Aty, Y., Franz, J., Mahmoud, M.A.W.: Bayesian prediction based on generalized order statistics using multiply Type-II censoring. *Statistics* **41**(6), 495–504 (2007)
175. Huang, S.-R., Wu, S.-J.: Bayesian estimation and prediction for Weibull model with progressive censoring. *J. Stat. Comput. Simul.* **82**(11), 1607–1620 (2012)
176. Ghafoori, S., Habibi Rad, A., Dostparast, M.: Bayesian two-sample prediction with progressively Type-II censored data for some lifetime models. *J. Iran. Stat. Soc. (JIRSS)* **10**, 63–86 (2011)
177. Wu, S.-J., Chen, D.-H., Chen, S.-T.: Bayesian inference for Rayleigh distribution under progressive censored sample. *Appl. Stoch. Models Bus. Ind.* **22**(3), 269–279 (2006)
178. Soliman, A.A., Al-Hossain, A.Y., Al-Harbi, M.M.: Predicting observables from Weibull model based on general progressive censored data with asymmetric loss. *Statist. Methodol.* **8**(5), 451–461 (2011)
179. Ali Mousa, M.A.M., Jaheen, Z.F.: Bayesian prediction for progressively censored data from the Burr model. *Statist. Papers* **43**(4), 587–593 (2002)
180. Jaheen, Z.F.: Prediction of progressive censored data from the Gompertz model. *Commun. Statist. Simulation Comput.* **32**(3), 663–676 (2003)
181. Klakattawi, H.S., Baharith, L.A., AL-Dayian, G.R.: Bayesian predictions of progressive censored data from the exponentiated modified Weibull distribution. *Canad. J. Comput. Math. Natur. Sci. Eng. Med.* **3**, 247–255 (2012)
182. Mohie El-Din, M.M., Shafay, A.R.: One- and two-sample Bayesian prediction intervals based on progressively Type-II censored data. *Statist. Papers* **54**, 287–307 (2013)
183. Ali Mousa, M.A.M., Al-Sagheer, S.: Bayesian prediction for progressively Type-II censored data from the Rayleigh model. *Commun. Statist. Theory Methods* **34**(12), 2353–2361 (2005)
184. Beutner, E., Cramer, E.: Using linear interpolation to reduce the order of the coverage error of nonparametric prediction intervals based on right-censored data. *J. Multivariate Anal.* **129**, 95–109 (2014)
185. Balakrishnan, N., Ng, H.K.T.: *Precedence-Type Tests and Applications*. Wiley, Hoboken (2006)

186. Ng, H.K.T., Balakrishnan, N.: Weighted precedence and maximal precedence tests and an extension to progressive censoring. *J. Statist. Plann. Inference* **135**(1), 197–221 (2005)
187. Balakrishnan, N., Tripathi, R.C., Kannan, N., Ng, H.K.T.: Some nonparametric precedence-type tests based on progressively censored samples and evaluation of power. *J. Statist. Plann. Inference* **140**(2), 559–573 (2010)
188. Alvarez-Andrade, S., Balakrishnan, N., Bordes, L.: Homogeneity tests based on several progressively Type-II censored samples. *J. Multivariate Anal.* **98**, 1195–1213 (2007)
189. Balakrishnan, N., Ng, H.K.T., Kannan, N.: A test of exponentiality based on spacings for progressively type-II censored data. In: Huber-Carol, C., Balakrishnan, N., Nikulin, M., Mesbah, M. (eds.) *Goodness-of-Fit Tests and Model Validity*, pp. 89–111. Birkhäuser, Boston (2002)
190. Marohn, F.: A characterization of generalized Pareto distributions by progressive censoring schemes and goodness-of-fit tests. *Commun. Statist. Theory Methods* **31**(7), 1055–1065 (2002)
191. Pakyari, R., Balakrishnan, N.: A general purpose approximate goodness-of-fit test for progressively Type-II censored data. *IEEE Trans. Reliab.* **61**, 238–244 (2012)
192. Burkschat, M., Cramer, E., Kamps, U.: On optimal schemes in progressive censoring. *Statist. Probab. Letters* **76**(10), 1032–1036 (2006)
193. Burkschat, M., Cramer, E., Kamps, U.: Optimality criteria and optimal schemes in progressive censoring. *Commun. Statist. Theory Methods* **36**, 1419–1431 (2007)
194. Salemi, U.H., Rezaei, S., Si, Y., Nadarajah, S.: On optimal progressive censoring schemes for normal distribution. *Annals of Data Science* **5**(4), 637–658 (2018)
195. Abo-Eleneen, Z.A.: A novel approach for optimal schemes in progressive censoring plans. *J. Commun. Comput.* **9**, 426–433 (2012)
196. Mishra, N.: Optimal one-step censoring schemes under entropy criterion. *Commun. Stat.- Simul. Comput.* **49**(8), 2068–2081 (2018)
197. Pradhan, B., Kundu, D.: On progressively censored generalized exponential distribution. *TEST* **18**, 497–515 (2009)
198. Pradhan, B., Kundu, D.: Inference and optimal censoring schemes for progressively censored Birnbaum-Saunders distribution. *J. Statist. Plann. Inference* **143**(0), 1098–1108 (2013)
199. Bhattacharya, R., Pradhan, B., Dewanji, A.: On optimum life-testing plans under Type-II progressive censoring scheme using variable neighborhood search algorithm. *TEST* **25**(2), 309–330 (2016)
200. Salemi, U.H., Rezaei, S., Nadarajah, S.: A-optimal and D-optimal censoring plans in progressively Type-II right censored order statistics. *Stat. Pap.* **60**(4), 1349–1367 (2017)
201. Cramer, E., Navarro, J.: Progressive Type-II censoring and coherent systems. *Nav. Res. Logist.* **62**, 512–530 (2015)
202. Cramer, E., Navarro, J.: The progressive censoring signature of coherent systems. *Appl. Stoch. Model. Bus. Ind.* **32**(5), 697–710 (2016)
203. Jablonka, A., Cramer, E., Hermanns, M.: Statistical inference for coherent systems with Weibull distributed component lifetimes under complete and incomplete information. *Appl. Stoch. Model. Bus. Ind.* **35**(4), 1011–1027 (2019)
204. Volterman, W., Balakrishnan, N.: Exact nonparametric confidence, prediction and tolerance intervals based on multi-sample Type-II right censored data. *J. Statist. Plann. Inference* **140**(11), 3306–3316 (2010)
205. Cohen, A.C.: Maximum likelihood estimation in the Weibull distribution based on complete and on censored samples. *Technometrics* **7**, 579–588 (1965)
206. Cohen, A.C.: Multi-censored sampling in the three parameter Weibull distribution. *Technometrics* **17**, 347–351 (1975)
207. Cohen, A.C.: Progressively censored sampling in the three parameter log-normal distribution. *Technometrics* **18**, 99–103 (1976)
208. Wingo, D.R.: Solution of the three-parameter Weibull equations by constrained modified quasilinearization (progressively censored samples). *IEEE Trans. Reliab.* **R-22**, 96–102 (1973)
209. Wingo, D.R.: Maximum likelihood methods for fitting the Burr type XII distribution to multiply (progressively) censored life test data. *Metrika* **40**, 203–210 (1993)
210. Nelson, W.: *Applied Life Data Analysis*. Wiley, New York (1982)
211. Cohen, A.C., Whitten, B.J.: *Parameter Estimation in Reliability and Life Span Models*. Marcel Dekker, New York (1988)
212. Balakrishnan, N., Han, D., Iliopoulos, G.: Exact inference for progressively Type-I censored exponential failure data. *Metrika* **73**, 335–358 (2011)
213. Balakrishnan, N., Iliopoulos, G.: Stochastic monotonicity of the MLE of exponential mean under different censoring schemes. *Ann. Inst. Statist. Math.* **61**, 753–772 (2009)
214. van Bentum, T., Cramer, E.: Stochastic monotonicity of MLEs of the mean for exponentially distributed lifetimes under sequential hybrid censoring. *Statist. Probab. Lett.* **148**, 1–8 (2019)
215. Casella, G., Berger, R.L.: *Statistical Inference*, 2 edn. Duxbury Press, Boston (2002)
216. Hahn, G.J., Meeker, W.Q., Escobar, L.A.: *Statistical Intervals: A Guide for Practitioners*. Wiley, New York (2017)
217. Balakrishnan, N., Cramer, E., Iliopoulos, G.: On the method of pivoting the CDF for exact confidence intervals with illustration for exponential mean under life-test with time constraints. *Statist. Probab. Lett.* **89**, 124–130 (2014)
218. Cramer, E., Górný, J., Laumen, B.: Multi-sample progressive Type-I censoring of exponentially distributed lifetimes. *Commun. Statist. Theory Methods* **50**(22), 5285–5313 (2020)
219. Cohen, A.C.: *Truncated and Censored Samples. Theory and Applications*. Marcel Dekker, New York (1991)
220. Cramer, E., Tamm, M.: On a correction of the scale MLE for a two-parameter exponential distribution under progressive Type-I censoring. *Commun. Statist. Theory Methods* **43**, 4401–4414 (2014)
221. Cohen, A.C.: Life testing and early failure. *Technometrics* **17**, 347–351 (1966)
222. Gibbons, D.I., Vance, L.C.: Estimators for the 2-parameter Weibull distribution with progressively censored samples. *IEEE Trans. Reliab.* **32**, 95–99 (1983)
223. Lemon, G.H.: Maximum likelihood estimation for the three parameter Weibull distribution based on censored samples. *Technometrics* **17**, 247–254 (1975)
224. Gajjar, A., Khatri, C.: Progressively censored samples from log-normal and logistic distributions. *Technometrics* **11**, 793–803 (1969)
225. Budhiraja, S., Pradhan, B., Sengupta, D.: Maximum likelihood estimators under progressive Type-I interval censoring. *Statist. Probab. Lett.* **123**, 202–209 (2017)
226. Cheng, C., Chen, J., Li, Z.: A new algorithm for maximum likelihood estimation with progressive Type-I interval censored data. *Commun. Statist. Simulation Comput.* **39**(4), 750–766 (2010)
227. Ng, H.K.T., Wang, Z.: Statistical estimation for the parameters of Weibull distribution based on progressively type-I interval censored sample. *J. Statist. Comp. Simul.* **79**(2), 145–159 (2009)
228. Lin, Y.-J., Lio, Y.L.: Bayesian inference under progressive type-I interval censoring. *J. Appl. Stat.* **39**(8), 1811–1824 (2012)
229. Chen, D.G., Lio, Y.L.: Parameter estimations for generalized exponential distribution under progressive Type-I interval censoring. *Comput. Statist. Data Anal.* **54**(6), 1581–1591 (2010)
230. Peng, X.-Y., Yan, Z.-Z.: Bayesian estimation for generalized exponential distribution based on progressive type-I interval censoring. *Acta Math. Appl. Sin. Engl. Ser.* **29**(2), 391–402 (2013)

231. Lio, Y.L., Chen, D.-G., Tsai, T.-R.: Parameter estimations for generalized Rayleigh distribution under progressively Type-I interval censored data. *Open J. Statist.* **1**(2), 46–57 (2011)
232. Xiuyun, P., Zaizai, Y.: Parameter estimations with gamma distribution based on progressive Type-I interval censoring. In: *IEEE International Conference on Computer Science and Automation Engineering (CSAE)*, 2011, pp. 449–453. IEEE, New York (2011)
233. Arabi Belaghi, R., Noori Asl, M., Singh, S.: On estimating the parameters of the Burr XII model under progressive Type-I interval censoring. *J. Stat. Comput. Simul.* **87**(16), 3132–3151 (2017)
234. Wu, S.-J., Chang, C.-T., Liao, K.-J., Huang, S.-R.: Planning of progressive group-censoring life tests with cost considerations. *J. Appl. Stat.* **35**(11), 1293–1304 (2008)
235. Lin, C.-T., Wu, S.J.S., Balakrishnan, N.: Planning life tests with progressively Type-I interval censored data from the lognormal distribution. *J. Statist. Plann. Inference* **139**(1), 54–61 (2009)
236. Lin, C.-T., Balakrishnan, N., Wu, S.J.S.: Planning life tests based on progressively Type-I grouped censored data from the Weibull distribution. *Commun. Statist. Simulation Comput.* **40**(4), 574–595 (2011)
237. Kuş, C., Akdoğan, Y., Wu, S.-J.: Optimal progressive group censoring scheme under cost considerations for pareto distribution. *J. Appl. Stat.* **40**(11), 2437–2450 (2013)
238. Singh, S., Tripathi, Y.M.: Estimating the parameters of an inverse Weibull distribution under progressive Type-I interval censoring. *Stat. Pap.* **59**, 21–56 (2018)
239. Roy, S., Pradhan, B.: Bayesian optimum life testing plans under progressive Type-I interval censoring scheme. *Qual. Reliab. Eng. Int.* **33**(8), 2727–2737 (2017)
240. Roy, S., Pradhan, B.: Bayesian C-optimal life testing plans under progressive type-I interval censoring scheme. *Appl. Math. Model.* **70**, 299–314 (2019)
241. Kaushik, A.: A progressive interval type-I censored life test plan for Rayleigh distribution. *Austrian Journal of Statistics* **48**(3), 76–86 (2019)
242. Balakrishnan, N., Kundu, D.: Hybrid censoring: Models, inferential results and applications (with discussions). *Comput. Statist. Data Anal.* **57**, 166–209 (2013)
243. Górný, J., Cramer, E.: From B-spline representations to gamma representations in hybrid censoring. *Stat. Pap.* **60**, 1119–1135 (2019)
244. Górný, J., Cramer, E.: A volume based approach to establish B-spline based expressions for density functions and its application to progressive hybrid censoring. *J. Korean Stat. Soc.* **38**, 340–355 (2019)
245. Burkschat, M., Cramer, E., Górný, J.: Type-I censored sequential k-out-of-n systems. *Appl. Math. Model.* **40**(19–20), 8156–8174 (2016)
246. Górný, J., Cramer, E.: Type-I hybrid censoring of multiple samples. *J. Comp. Appl. Math.* **366**, 112404 (2020)
247. Chan, P.S., Ng, H.K.T., Su, F.: Exact likelihood inference for the two-parameter exponential distribution under Type-II progressively hybrid censoring. *Metrika* **78**(6), 1–24 (2015)
248. Lin, C.-T., Chou, C.-C., Huang, Y.-L.: Inference for the Weibull distribution with progressive hybrid censoring. *Comp. Statist. Data Anal.* **56**(3), 451–467 (2012)
249. Golparvar, L., Parsian, A.: Inference on proportional hazard rate model parameter under Type-I progressively hybrid censoring scheme. *Commun. Stat.- Theory Methods* **45**(24), 7258–7274 (2016)
250. Hemmati, F., Khorram, E.: Statistical analysis of the log-normal distribution under Type-II progressive hybrid censoring schemes. *Commun. Statist. Simulation Comput.* **42**(1), 52–75 (2013)
251. Tomer, S.K., Panwar, M.S.: Estimation procedures for Maxwell distribution under type-I progressive hybrid censoring scheme. *J. Statist. Comp. Simul.* **85**, 339–356 (2014)
252. Kayal, T., Tripathi, Y.M., Rastogi, M.K., Asgharzadeh, A.: Inference for Burr XII distribution under Type I progressive hybrid censoring. *Commun. Stat.- Simul. Comput.* **46**(9), 7447–7465 (2017)
253. Sen, T., Singh, S., Tripathi, Y.M.: Statistical Inference for Lognormal Distribution with Type-I Progressive Hybrid Censored Data. *Am. J. Math. Manag. Sci.* **38**(1), 70–95 (2019)
254. Arabi Belaghi, R., Noori Asl, M.: Estimation based on progressively type-I hybrid censored data from the Burr XII distribution. *Stat. Pap.* **60**, 411–453 (2019)
255. Górný, J., Cramer, E.: Type-I hybrid censoring of uniformly distributed lifetimes. *Commun. Stat.- Theory Methods* **48**, 412–433 (2019)
256. Bhattacharya, R., Pradhan, B.: Computation of optimum Type-II progressively hybrid censoring schemes using variable neighborhood search algorithm. *TEST* **26**(4), 802–821 (2017)
257. Bhattacharya, R., Saha, B.N., Farias, G.G., Balakrishnan, N.: Multi-criteria-based optimal life-testing plans under hybrid censoring scheme. *TEST* **29**(2), 430–453 (2020)
258. Cramer, E., Burkschat, M., Górný, J.: On the exact distribution of the MLEs based on Type-II progressively hybrid censored data from exponential distributions. *J. Stat. Comp. Simul.* **86**, 2036–2052 (2016)
259. Jansen, M., Górný, J., Cramer, E.: Exact likelihood inference for an exponential parameter under a multi-sample Type-II progressive hybrid censoring model. submitted for publication. *Am. J. Math. Manag. Sci.* **41**, 101–127 (2022)
260. Lin, C.-T., Ng, H.K.T., Chan, P.S.: Statistical inference of Type-II progressively hybrid censored data with Weibull lifetimes. *Commun. Statist. Theory Methods* **38**(10), 1710–1729 (2009)
261. Mokhtari, E.B., Rad, A.H., Yousefzadeh, F.: Inference for Weibull distribution based on progressively Type-II hybrid censored data. *J. Statist. Plann. Inference* **141**(8), 2824–2838 (2011)
262. Alma, O.G., Arabi Belaghi, R.: On the estimation of the extreme value and normal distribution parameters based on progressive type-II hybrid-censored data. *J. Stat. Comput. Simul.* **86**(3), 569–596 (2016)
263. Noori Asl, M., Arabi Belaghi, R., Bevrani, H.: On Burr XII distribution analysis under progressive type-II hybrid censored data. *Methodol. Comput. Appl. Probab.* **19**(2), 665–683 (2017)
264. Cho, Y., Sun, H., Lee, K.: Exact likelihood inference for an exponential parameter under generalized progressive hybrid censoring scheme. *Statist. Methodol.* **23**, 18–34 (2015)
265. Cho, Y., Sun, H., Lee, K.: Estimating the entropy of a Weibull distribution under generalized progressive hybrid censoring. *Entropy* **17**, 102–122 (2015)
266. Górný, J., Cramer, E.: Exact likelihood inference for exponential distributions under generalized progressive hybrid censoring schemes. *Stat. Methodol.* **29**, 70–94 (2016)
267. Lee, K., Sun, H., Cho, Y.: Exact likelihood inference of the exponential parameter under generalized Type II progressive hybrid censoring. *J. Korean Stat. Soc.* **45**(1), 123–136 (2016)
268. Seo, J.I., Kim, Y.: Robust Bayesian estimation of a two-parameter exponential distribution under generalized Type-I progressive hybrid censoring. *Commun. Stat. Simul. Comput.* **46**(7), 5795–5807 (2017)
269. Seo, J.I., Kim, Y.: Robust Bayesian analysis for exponential parameters under generalized Type-II progressive hybrid censoring. *Commun. Stat.- Theory Methods* **47**(9), 2259–2277 (2018)
270. Mohie El-Din, M.M., Nagy, M., Abu-Moussa, M.H.: Estimation and prediction for Gompertz distribution under the generalized progressive hybrid censored data. *Annals of Data Science* **6**, 673–705 (2019)
271. Górný, J., Cramer, E.: Exact inference for a new flexible hybrid censoring scheme. *Journal of the Indian Society for Probability and Statistics* **19**(1), 169–199 (2018)

272. Park, S., Balakrishnan, N., Kim, S.W.: Fisher information in progressive hybrid censoring schemes. *Statistics* **45**, 623–631 (2011)
273. Sen, T., Pradhan, B., Tripathi, Y.M., Bhattacharya, R.: Fisher information in generalized progressive hybrid-censored data. *Statistics* **52**(5), 1025–1039 (2018)
274. Kohansal, A., Nadarajah, S.: Stress–strength parameter estimation based on type-II hybrid progressive censored samples for a Kumaraswamy distribution. *IEEE Trans. Reliab.* **68**, 1296–1310 (2019)
275. Lin, C.-T., Huang, Y.-L.: On progressive hybrid censored exponential distribution. *J. Stat. Comput. Simul.* **82**, 689–709 (2012)
276. Sobhi, M.M.A., Soliman, A.A.: Estimation for the exponentiated Weibull model with adaptive Type-II progressive censored schemes. *Appl. Math. Model.* **40**(2), 1180–1192 (2016)
277. Yuen, H.-K., Tse, S.-K.: Parameters estimation for Weibull distributed lifetimes under progressive censoring with random removals. *J. Stat. Comput. Simul.* **55**(1-2), 57–71 (1996)
278. Tse, S.-K., Yang, C., Yuen, H.-K.: Statistical analysis of Weibull distributed lifetime data under Type II progressive censoring with binomial removals. *J. Appl. Stat.* **27**(8), 1033–1043 (2000)
279. Tse, S.-K., Xiang, L.: Interval estimation for Weibull-distributed life data under Type II progressive censoring with random removals. *J. Biopharm. Stat.* **13**, 1–16 (2003)
280. Xiang, L., Tse, S.K.: Maximum likelihood estimation in survival studies under progressive interval censoring with random removals. *J. Biopharm. Stat.* **15**(6), 981–991 (2005)
281. Kaushik, A., Singh, U., Singh, S.K.: Bayesian inference for the parameters of Weibull distribution under progressive Type-I interval censored data with beta-binomial removals. *Commun. Stat. Simul. Comput.* **46**(4), 3140–3158 (2017)
282. Budhiraja, S., Pradhan, B.: Point and interval estimation under progressive type-I interval censoring with random removal. *Statistical papers* **61**(1), 445–477 (2020)
283. Bairamov, I., Parsi, S.: On flexible progressive censoring. *J. Comput. Appl. Math.* **235**(16), 4537–4544 (2011)
284. Kinaci, I.: A generalization of flexible progressive censoring. *Pakistan J. Stat.* **29**, 377–387 (2013)
285. Park, S., Ng, H.K.T., Chan, P.S.: On the Fisher information and design of a flexible progressive censored experiment. *Statist. Probab. Lett.* **97**, 142–149 (2015)
286. Li, F.: Reliability analysis for the Pareto model based on the progressive type II censored sample. In: Jin, D., Lin, S. (eds.) *Advances in Computer Science, Intelligent System and Environment*, pp. 467–472. Springer, Berlin (2011)
287. Krishna, H., Kumar, K.: Reliability estimation in generalized inverted exponential distribution with progressively type II censored sample. *J. Stat. Comput. Simul.* **83**, 1007–1019 (2013)
288. Rastogi, M.K., Tripathi, Y.M.: Parameter and reliability estimation for an exponentiated half-logistic distribution under progressive type II censoring. *J. Stat. Comput. Simul.* **84**, 1711–1727 (2014)
289. Guo, L., Gui, W.: Statistical inference of the reliability for generalized exponential distribution under progressive Type-II censoring schemes. *IEEE Trans. Reliab.* **67**(2), 470–480 (2018)
290. Maiti, K., Kayal, S.: Estimation of parameters and reliability characteristics for a generalized Rayleigh distribution under progressive type-II censored sample. *Commun. Stat. Simul. Comput.* **50**(11), 3669–3698 (2021)
291. Zhang, Z., Gui, W.: Statistical inference of reliability of Generalized Rayleigh distribution under progressively type-II censoring. *J. Comput. Appl. Math.* **361**, 295–312 (2019)
292. Asgharzadeh, A., Valiollahi, R.: Inference for the proportional hazards family under progressive Type-II censoring. *J. Iran. Stat. Soc. (JIRSS)* **8**, 35–53 (2009)
293. Chaturvedi, A., Kumar, N., Kumar, K.: Statistical inference for the reliability functions of a family of lifetime distributions based on progressive type II right censoring. *Statistica* **78**(1), 81–101 (2018)
294. Kotz, S., Lumelskii, Y., Pinsky, M.: *The Stress-Strength Model and Its Generalizations—Theory and Applications*. World Scientific Publications, Singapore (2003)
295. Cramer, E., Kamps, U.: The UMVUE of $P(X < Y)$ based on Type-II censored samples from Weibull multivariate exponential distributions. *Metrika* **46**, 93–121 (1997)
296. Cramer, E.: Inference for stress-strength systems based on Weibull multivariate exponential samples. *Commun. Statist. Theory Methods* **30**, 331–346 (2001)
297. Saraçoğlu, B., Kinaci, I., Kundu, D.: On estimation of $R = P(Y < X)$ for exponential distribution under progressive type-II censoring. *J. Stat. Comput. Simul.* **82**, 729–744 (2012)
298. Krishna, H., Dube, M., Garg, R.: Estimation of $P(Y < X)$ for progressively first-failure-censored generalized inverted exponential distribution. *J. Stat. Comput. Simul.* **87**(11), 2274–2289 (2017)
299. Asgharzadeh, A., Valiollahi, R., Raqab, M.Z.: Stress-strength reliability of Weibull distribution based on progressively censored samples. *Statist. Oper. Res. Trans.* **35**, 103–124 (2011)
300. Valiollahi, R., Asgharzadeh, A., Raqab, M.Z.: Estimation of $P(Y < X)$ for Weibull distribution under progressive Type-II censoring. *Commun. Statist. Theory Methods* **42**(24), 4476–4498 (2013)
301. Rezaei, S., Noughabi, R.A., Nadarajah, S.: Estimation of stress-strength reliability for the generalized Pareto distribution based on progressively censored samples. *Annals of Data Science* **2**(1), 83–101 (2015)
302. Basirat, M., Baratpour, S., Ahmadi, J.: Statistical inferences for stress-strength in the proportional hazard models based on progressive Type-II censored samples. *J. Stat. Comput. Simul.* **85**, 431–449 (2014)
303. Bai, X., Shi, Y., Liu, Y., Liu, B.: Reliability inference of stress-strength model for the truncated proportional hazard rate distribution under progressively Type-II censored samples. *Appl. Math. Model.* **65**, 377–389 (2019)
304. Xiuyun, P., Yan, X., Zaizai, Y.: Reliability analysis of Birnbaum–Saunders model based on progressive Type-II censoring. *J. Stat. Comput. Simul.* **89**(3), 461–477 (2019)
305. Babayi, S., Khorram, E.: Inference of stress-strength for the Type-II generalized logistic distribution under progressively Type-II censored samples. *Commun. Stat. Simul. Comput.* **47**(7), 1975–1995 (2018)
306. Bai, X., Shi, Y., Liu, Y., Liu, B.: Reliability estimation of stress-strength model using finite mixture distributions under progressively interval censoring. *J. Comput. Appl. Math.* **348**, 509–524 (2019)
307. Lin, C.-T., Ke, S.-J.: Estimation of $P(Y < X)$ for location-scale distributions under joint progressively type-II right censoring. *Quality Technology & Quantitative Management* **10**(3), 339–352 (2013)
308. Kundu, D., Kannan, N., Balakrishnan, N.: Analysis of progressively censored competing risks data. In: Balakrishnan, N., Rao, C.R. (eds.) *Advances in Survival Analysis. Handbook of Statistics*, vol. 23, pp. 331–348. Elsevier, Amsterdam (2004)
309. Pareek, B., Kundu, D., Kumar, S.: On progressively censored competing risks data for Weibull distributions. *Comput. Statist. Data Anal.* **53**, 4083–4094 (2009)
310. Kundu, D., Pradhan, B.: Bayesian analysis of progressively censored competing risks data. *Sankhyā B* **73**(2), 276–296 (2011)
311. Chacko, M., Mohan, R.: Bayesian analysis of Weibull distribution based on progressive Type-II censored competing risks data with binomial removals. *Comput. Stat.* **34**(1), 233–252 (2019)
312. Cramer, E., Schmiedt, A.B.: Progressively Type-II censored competing risks data from Lomax distributions. *Comp. Statist. Data Anal.* **55**, 1285–1303 (2011)

313. AL-Hussaini, E.K., Abdel-Hamid, A.H., Hashem, A.F.: Bayesian prediction intervals of order statistics based on progressively Type-II censored competing risks data from the half-logistic distribution. *J. Egypt. Math. Soc.* **23**(1), 190–196 (2015)
314. Wang, L.: Inference of progressively censored competing risks data from Kumaraswamy distributions. *J. Comput. Appl. Math.* **343**, 719–736 (2018)
315. Azizi, F., Haghghi, F., Gilani, N.T.: Statistical inference for competing risks model under progressive interval censored Weibull data. *Commun. Stat. Simul. Comput.* **49**(7), 1–14 (2018)
316. Ahmadi, K., Yousefzadeh, F., Rezaei, M.: Analysis of progressively type-I interval censored competing risks data for a class of an exponential distribution. *J. Stat. Comput. Simul.* **86**(18), 3629–3652 (2016)
317. Kundu, D., Gupta, R.D.: Analysis of hybrid life-tests in presence of competing risks. *Metrika* **65**(2), 159–170 (2006)
318. Kundu, D., Joarder, A.: Analysis of Type-II progressively hybrid censored competing risks data. *J. Mod. Appl. Statist. Methods* **5**(1), 152–170 (2006)
319. Mao, S., Shi, Y.-M., Sun, Y.-D.: Exact inference for competing risks model with generalized Type-I hybrid censored exponential data. *J. Stat. Comp. Simul.* **84**(11), 2506–2521 (2014)
320. Bhattacharya, S., Pradhan, B., Kundu, D.: Analysis of hybrid censored competing risks data. *Statistics* **48**(5), 1138–1154 (2014)
321. Feizjavadian, S., Hashemi, R.: Analysis of dependent competing risks in the presence of progressive hybrid censoring using Marshall–Olkin bivariate Weibull distribution. *Comput. Stat. Data Anal.* **82**, 19–34 (2015)
322. Iliopoulos, G.: On exact confidence intervals in a competing risks model with generalized hybrid type-I censored exponential data. *J. Stat. Comp. Simul.* **85**, 2953–2961 (2015)
323. Koley, A., Kundu, D.: On generalized progressive hybrid censoring in presence of competing risks. *Metrika* **80**, 401–426 (2017)
324. Koley, A., Kundu, D., Ganguly, A.: Analysis of Type-II hybrid censored competing risks data. *Statistics* **51**(6), 1304–1325 (2017)
325. Wang, L.: Inference for Weibull competing risks data under generalized progressive hybrid censoring. *IEEE Trans. Reliab.* **67**(3), 998–1007 (2018)
326. Wu, S.-J., Kuş, C.: On estimation based on progressive first-failure-censored sampling. *Comp. Statist. Data Anal.* **53**(10), 3659–3670 (2009)
327. Soliman, A.A., Abd-Ellah, A.H., Abou-Elheggag, N.A., Abd-Elmougod, G.A.: Estimation of the parameters of life for Gompertz distribution using progressive first-failure censored data. *Comp. Statist. Data Anal.* **56**(8), 2471–2485 (2012)
328. Potdar, K.G., Shirke, D.T.: Inference for the scale parameter of lifetime distribution of k -unit parallel system based on progressively censored data. *J. Stat. Comput. Simul.* **84**, 171–185 (2013)
329. Hermanns, M., Cramer, E.: Likelihood inference for the component lifetime distribution based on progressively censored parallel systems data. *J. Stat. Comput. Simul.* **87**, 607–630 (2017)
330. Amiri, L., Ganjali, M., Hashemi, R., Khazaei, M.: The competing risks analysis for parallel and series systems using Type-II progressive censoring. *Commun. Stat.- Theory Methods* **49**(22) (2019)
331. Hermanns, M., Cramer, E.: Inference with progressively censored k -out-of- n system lifetime data. *TEST* **27**, 787–810 (2018)
332. Hermanns, M., Cramer, E., Ng, H.K.T.: EM algorithms for ordered and censored system lifetime data under a proportional hazard rate model. *J. Stat. Comput. Simul.* **90**(18), 3301–3337 (2020)
333. Balasooriya, U., Saw, S.L.C.: Reliability sampling plans for the two-parameter exponential distribution under progressive censoring. *J. Appl. Stat.* **25**, 707–714 (1998)
334. Fernández, A.J.: Progressively censored variables sampling plans for two-parameter exponential distributions. *J. Appl. Stat.* **32**(8), 823–829 (2005)
335. Pérez-González, C., Fernández, A.J.: Accuracy of approximate progressively censored reliability sampling plans for exponential models. *Statist. Papers* **50**, 161–170 (2009)
336. Fernández, A.J., Pérez-González, C.J., Aslam, M., Jun, C.-H.: Design of progressively censored group sampling plans for Weibull distributions: An optimization problem. *Eur. J. Oper. Res.* **211**(3), 525–532 (2011)
337. Balasooriya, U., Balakrishnan, N.: Reliability sampling plans for the lognormal distribution, based on progressively censored samples. *IEEE Trans. Reliab.* **49**, 199–203 (2000)
338. Wu, S.-J., Huang, S.-R.: Progressively first-failure censored reliability sampling plans with cost constraint. *Comp. Statist. Data Anal.* **56**(6), 2018–2030 (2012)
339. Laumen, B., Cramer, E.: Inference for the lifetime performance index with progressively Type-II censored samples from gamma distributions. *Econom. Quality Control* **30**, 59–73 (2015)
340. Lee, W.-C., Wu, J.-W., Hong, C.-W.: Assessing the lifetime performance index of products with the exponential distribution under progressively type II right censored samples. *J. Comput. Appl. Math.* **231**(2), 648–656 (2009)
341. Lee, W.-C., Wu, J.-W., Hong, M.-L., Lin, L.-S., Chan, R.-L.: Assessing the lifetime performance index of Rayleigh products based on the Bayesian estimation under progressive Type II right censored samples. *J. Comput. Appl. Math.* **235**(6), 1676–1688 (2011)
342. Lee, W.-C., Wu, J.-W., Hong, C.-W.: Assessing the lifetime performance index of products from progressively Type II right censored data using Burr XII model. *Math. Comput. Simulation* **79**(7), 2167–2179 (2009)
343. Ahmadi, M.V., Doostparast, M., Ahmadi, J.: Statistical inference for the lifetime performance index based on generalised order statistics from exponential distribution. *Int. J. System Sci.* **46**, 1094–1107 (2014)
344. Ahmadi, M.V., Doostparast, M., Ahmadi, J.: Estimating the lifetime performance index with Weibull distribution based on progressive first-failure censoring scheme. *J. Comput. Appl. Math.* **239**, 93–102 (2013)
345. Hong, C.W., Lee, W.C., Wu, J.W.: Computational procedure of performance assessment of lifetime index of products for the Weibull distribution with the progressive first-failure-censored sampling plan. *J. Appl. Math.* **2012**(Article ID 717184), 13 pages (2012)
346. Mahmoud, M.A.W., El-Sagheer, R.M., Soliman, A.A.-E., Ellah, A.H.A.: Inferences of the lifetime performance index with Lomax distribution based on progressive type-II censored data. *Econom. Quality Control* **29**, 39–51 (2014)
347. Ahmadi, M.V., Doostparast, M.: Pareto analysis for the lifetime performance index of products on the basis of progressively first-failure-censored batches under balanced symmetric and asymmetric loss functions. *J. Appl. Stat.* **46**(7), 1196–1227 (2019)
348. Wu, S.-F., Hsieh, Y.-T.: The assessment on the lifetime performance index of products with Gompertz distribution based on the progressive type I interval censored sample. *J. Comput. Appl. Math.* **351**, 66–76 (2019)
349. Wu, S.-F., Lin, Y.-P.: Computational testing algorithmic procedure of assessment for lifetime performance index of products with one-parameter exponential distribution under progressive type I interval censoring. *Math. Comput. Simul.* **120**, 79–90 (2016)
350. Wu, S.-F., Lin, M.-J.: Computational testing algorithmic procedure of assessment for lifetime performance index of products with Weibull distribution under progressive type I interval censoring. *J. Comput. Appl. Math.* **311**, 364–374 (2017)

351. Kundu, D., Ganguly, A.: Analysis of Step-Stress Models. Academic Press Inc., London (2017)
352. Balakrishnan, N.: A synthesis of exact inferential results for exponential step-stress models and associated optimal accelerated life-tests. *Metrika* **69**, 351–396 (2009)
353. Xie, Q., Balakrishnan, N., Han, D.: Exact inference and optimal censoring scheme for a simple step-stress model under progressive Type-II censoring. In: Balakrishnan, N. (ed.) *Advances in Mathematical and Statistical Modeling*, pp. 107–137. Birkhäuser, Boston (2008)
354. Balakrishnan, N., Xie, Q., Kundu, D.: Exact inference for a simple step-stress model from the exponential distribution under time constraint. *Ann. Inst. Statist. Math.* **61**(1), 251–274 (2009)
355. Liu, F., Shi, Y.: Inference for a simple step-stress model with progressively censored competing risks data from Weibull distribution. *Commun. Stat.- Theory Methods* **46**(14), 7238–7255 (2017)
356. Gouno, E., Sen, A., Balakrishnan, N.: Optimal step-stress test under progressive Type-I censoring. *IEEE Trans. Reliab.* **53**, 388–393 (2004)
357. Han, D., Balakrishnan, N., Sen, A., Gouno, E.: Corrections on 'Optimal step-stress test under progressive Type-I censoring'. *IEEE Trans. Reliab.* **55**, 613–614 (2006)
358. Wu, S.-J., Lin, Y.-P., Chen, Y.-J.: Planning step-stress life test with progressively type I group-censored exponential data. *Stat. Neerl.* **60**(1), 46–56 (2006)
359. Wu, S.-J., Lin, Y.-P., Chen, S.-T.: Optimal step-stress test under type I progressive group-censoring with random removals. *J. Statist. Plann. Inference* **138**(4), 817–826 (2008)
360. Yue, H.-B., Shi, Y.-M.: Optimal sample size allocation for multi-level stress testing under progressive hybrid interval censoring. *Appl. Mech. Materials* **423–426**, 2423–2426 (2013)
361. Wang, B.X., Yu, K.: Optimum plan for step-stress model with progressive Type-II censoring. *TEST* **18**, 115–135 (2009)
362. Balakrishnan, N., Kamps, U., Kateri, M.: A sequential order statistics approach to step-stress testing. *Ann. Inst. Statist. Math.* **64**, 303–318 (2012)
363. Wang, B.X.: Interval estimation for exponential progressive Type-II censored step-stress accelerated life-testing. *J. Statist. Plann. Inference* **140**(9), 2706–2718 (2010)
364. Abdel-Hamid, A.H., AL-Hussaini, E.K.: Inference for a progressive stress model from Weibull distribution under progressive type-II censoring. *J. Comput. Appl. Math.* **235**(17), 5259–5271 (2011)
365. Balakrishnan, N., Cramer, E., Dembińska, A.: Characterizations of geometric distribution through progressively Type-II right censored order statistics. *Statistics* **59**, 559–573 (2011)
366. Laumen, B., Cramer, E.: Stage life testing. *Nav. Res. Logistics* **53**, 632–647 (2019)
367. Laumen, B., Cramer, E.: k -step stage life testing. *Stat. Neerlandica* **75**, 203–223 (2021)
368. Laumen, B., Cramer, E.: Stage life testing with random stage changing times. *Commun. Stat.- Theory Methods* **51**, 3934–3959 (2022)
369. Laumen, B.: *Progressive Censoring and Stage Life Testing* Ph.D. Thesis. RWTH Aachen University, Aachen (2017)
370. Rasouli, A., Balakrishnan, N.: Exact likelihood inference for two exponential populations under joint progressive Type-II censoring. *Commun. Statist. Theory Methods* **39**(12), 2172–2191 (2010)
371. Parsi, S., Bairamov, I.: Expected values of the number of failures for two populations under joint Type-II progressive censoring. *Comp. Statist. Data Anal.* **53**(10), 3560–3570 (2009)
372. Parsi, S., Ganjali, M., Farsipour, N.S.: Conditional maximum likelihood and interval estimation for two Weibull populations under joint Type-II progressive censoring. *Commun. Statist. Theory Methods* **40**(12), 2117–2135 (2011)
373. Doostparast, M., Ahmadi, M.V., Ahmadi, J.: Bayes estimation based on joint progressive type II censored data under LINEX loss function. *Commun. Stat. Simul. Comput.* **42**(8), 1865–1886 (2013)
374. Balakrishnan, N., Su, F., Liu, K.-Y.: Exact likelihood inference for k exponential populations under joint progressive type-II censoring. *Commun. Stat. Simul. Comput.* **44**(4), 902–923 (2015)
375. Mondal, S., Kundu, D.: Point and interval estimation of Weibull parameters based on joint progressively censored data. *Sankhya B* **81**, 1–25 (2019)
376. Mondal, S., Kundu, D.: On the joint Type-II progressive censoring scheme. *Commun. Stat.- Theory Methods* **49**, 958–976 (2020)
377. Ashour, S.K., Abo-Kasem, O.E.: Statistical inference for two exponential populations under joint progressive Type-I censored scheme. *Commun. Stat.- Theory Methods* **46**(7), 3479–3488 (2017)
378. Abo-Kasem, O.E., Nassar, M., Dey, S., Rasouli, A.: Classical and Bayesian estimation for two exponential populations based on joint type-I progressive hybrid censoring scheme. *Am. J. Math. Manag. Sci.* **38**, 373–385 (2019)
379. Mondal, S., Kundu, D.: Bayesian inference for Weibull distribution under the balanced joint Type-II progressive censoring scheme. *Am. J. Math. Manag. Sci.* **39**(1), 56–74 (2019)
380. Balakrishnan, N., Cramer, E., Kundu, D.: Hybrid censoring know-how. In: *Designs and Implementations*. Academic Press (2023)



Narayanaswamy Balakrishnan received his B.Sc. and M.Sc. degrees in Statistics from the University of Madras, India, in 1976 and 1978, respectively. He finished his Ph.D. in Statistics from the Indian Institute of Technology, Kanpur, India, in 1981. He is a Distinguished University Professor at McMaster University, Hamilton, ON, Canada. His research interests include distribution theory, ordered data analysis, censoring methodology, reliability, survival analysis, non-parametric inference, and statistical quality control. Prof. Balakrishnan is a Fellow of the American Statistical Association and a Fellow of the Institute of Mathematical Statistics. He is currently the Editor-in-Chief of Communications in Statistics. He has received an honorary doctorate degree from The National University of Athens in 2016 for his valuable contributions to the field of Statistics.



Erhard Cramer received his Ph.D. from RWTH Aachen University in 1997. He worked did research and taught at University of Oldenburg and at TU Darmstadt before he was appointed Professor of Applied Probability at RWTH Aachen University in 2004. He works now on statistical and probabilistic analysis of ordered data, censoring models, lifetime data, and reliability. He currently serves as associate editor of several statistical journals including Computational Statistics and Communications in Statistics.



Contents

10.1 Introduction	185
10.2 Classification of Warranty Policies	186
10.2.1 Renewable and Nonrenewable Warranties	186
10.2.2 FRW, FRPW, PRW, CMW, and FSW Policies	187
10.2.3 Repair-Limit Warranty	188
10.2.4 One-Attribute Warranty and Two-Attribute Warranty . . .	189
10.3 Evaluation of Warranty Policies	189
10.3.1 Warranty Cost Factors	189
10.3.2 Criteria for Comparison of Warranties	190
10.3.3 Warranty Cost Evaluation for Complex Systems	191
10.3.4 Assessing Warranty Benefits	192
10.3.5 On the Optimal Warranty Policy	192
10.4 Concluding Remarks	193
References	194

Abstract

Warranty is a topic that has been studied extensively by different disciplines including engineering, economics, management science, accounting, and marketing researchers (Blischke and Murthy, Warranty cost analysis. Marcel Dekker, New York, 1994, p 47). Warranty policy is a guarantee for the seller of a product to provide the buyer with a specific service, such as replacement or repair, in the event of the product failure. Today warranty policy is an important marketing factor used by the manufacturers and corporates to promote its product to consumers (Park and Pham, IEEE Trans Syst Man Cybern A

40:1329–1340, 2010). This chapter aims to provide an overview on warranties focusing on the cost and benefit perspective of various warranty and maintenance policies. After a brief introduction of the current status of warranty research, the second part of this chapter classifies various existing and several recent promotional warranty policies to extend the taxonomy initiated by *Blischke and Murthy* (Eur J Oper Res 62:127–148, 1993).

Focusing on the quantitative modeling perspective of both the cost and benefit analyses of warranties, we summarize five problems that are essential to warranty issuers. These problems are: (i) what are the warranty cost factors; (ii) how to compare different warranty policies; (iii) how to analyze the warranty cost of multi-component systems; (iv) how to evaluate the warranty benefits; (v) how to determine the optimal warranty policy.

A list of future warranty research topics are presented in the last part of this chapter. We hope that this will stimulate further interest among researchers and practitioners.

Keywords

Preventive maintenance · Warranty period · Warranty service · Minimal repair · Indifference price · Warrant policies

10.1 Introduction

Warranty is an obligation attached to products (items or systems) that requires the warranty issuers (manufacturers or suppliers) to provide compensation to consumers according to the warranty terms when the warranted products fail to perform their pre-specified functions under normal usage within the warranty coverage period. Similar definitions can be found in *Blischke and Murthy* [1, 2], *McGuire* [3], and *Singpurwalla and Wilson et al.* [4]. Based on this definition, a warranty contract should contain at least three

H. Pham (✉)
Department of Industrial and Systems Engineering, Rutgers
University, Piscataway, NJ, USA
e-mail: hopham@soe.rutgers.edu

J. Bai
Barclays Bank US, Wilmington, DE, USA

characteristics: the coverage period (fixed or random), the method of compensations, and the conditions under which such compensations would be offered. The last characteristic is closely related to warranty execution since it clarifies consumers, rights and protects warranty issuers from excessive false claims. From the costing perspective, the first two characteristics are more important to manufacturers because they determine the depth of the protection against premature failures and the direct cost related to those failures.

Traditionally, warranty serves as a protection instrument attached to products sold to consumers. There are two facets of the protection role: on one hand, it guarantees a properly functioning product for at least a period of w , either financially or physically. On the other hand, it also specifies an upper bound on the liability of the supplier induced by the warranty. In addition to the protection role, warranty has always been one of the most important elements in business marketing strategy. As indicated in [3, p. 1], manufacturers, primary rationale for offering warranty is to support their products to gain some advantage in the market, either by expressing the company's faith in the product quality or by competing with other firms. Due to the more than ever fierce competition in the modern economy, the market promotion role of warranty has become even more significant. Manufacturers are fighting with each other through various channels from competitive pricing, improved product reliability, to more comprehensive warranties. Because of technology constraints or time constraint, it is usually difficult to improve product quality in a short time. As a result, warranty has evolved as an essential part of marketing strategy, along with pricing and advertising, which is especially powerful during the introduction period of new, expensive products such as automobiles and complex machinery.

In the last two decades, warranty has been studied extensively among many disciplines such as engineering, economics, statistics, marketing and management science, to name a few. Consequently, the literature on warranty is not only vast, but also disjoint [1]. There are three books and hundreds of journal articles that have addressed warranty-related problems within the last 10 years. A comprehensive collection of related references up to 1996 can be found in [2]. In general, researchers in engineering are interested in quality control and improving product reliability to reduce production and service costs. Some of the major references are Chen et al. [5], Djamaludin et al. [6], Hedge and Kubat [7], Mi [8], Murthy and Hussain [9], Nguyen and Murthy [10], and Sahin [11]. Economists usually treat warranty as a special type of insurance. Consequently, they developed the economic theory of warranties as one of many applications of microeconomics. We refer readers to DeCroix [12], Emons [13, 14], Lutz and Padmanabhan [15], Padmanabhan and Rao [16], Murthy and Asgharizadeh [17] and the references therein. Statisticians mainly focus on warranty

claim prediction, statistical inference of warranty cost, and estimation of product reliability or availability. Some of the key references are Frees [18, 19], Ja et al. [20], Kalbfleisch [21], Kao and Smith [22, 23], Menzefricke [24], Padmanabhan and Worm [25] and Polatoglu [26]. A long-term trend in warranty study is the focus on various warranty-management aspects. Some recent references are Chun and Tang [27], Ja et al. [20], Lam and Lam [28], Wang and Sheu [29], and Yeh et al. [30, 31]. Blischke and Murthy [32] developed a framework for the analytical study of various issues related to warranty. Recently, Murthy and Djamaludin [33] enriched the framework by summarizing the literature since 1992 from an overall business perspective. Another review by Thomas and Rao [34] provided some suggestions for expanding the analysis methods for making warranty decisions. Park and Pham [35] discussed several cost models with consideration of non-renewable and renewable warranty policies based on two-dimension aspects such as failure times and repair times.

In this chapter, we briefly review some recent work in warranty literature from the manufacturers, perspective. The objectives of this chapter are to classify various existing and relatively new warranty policies to extend the taxonomy discussed in [36], and to summarize and illustrate some fundamental warranty economic problems including several warranty-maintenance policies such as the block replacement and age replacement policies.

10.2 Classification of Warranty Policies

Numerous warranty policies have been studied in the last several decades. Blischke and Murthy [37] presented a taxonomy of more than 18 warranty policies and provided a precise statement of each of them. In this section, we extend the taxonomy by addressing several recently proposed policies that might be of interests to warranty managers. It should be noted that we mainly focus on type A policies [37], which, based on the taxonomy, are referred to as policies for single items and not involving product development.

10.2.1 Renewable and Nonrenewable Warranties

One of the basic characteristics of warranties is whether they are renewable or not. For a regular renewable policy with warranty period w , whenever a product fails within w , the buyer is compensated according to the terms of the warranty contract and the warranty policy is renewed for another period w . As a result, a warranty cycle T , starting from the date of sale, ending at the warranty expiration date, is a random variable whose value depends on w , the total number of failures under the warranty, and the actual failure

inter-arrival times. Renewable warranties are often offered for inexpensive, nonrepairable consumer electronic products such as microwaves, coffee makers, and so forth, either implicitly or explicitly. One should notice that theoretically the warranty cycle for a renewable policy can be arbitrarily large. For example, consumers can induce the failures so that they keep on getting new warranties indefinitely. Such moral hazard problems might be one of the reasons that renewable policies are not as popular as nonrenewable ones among warranty issuers.

One way to remedy this problem is to modify the regular renewable policy in the following way: instead of offering the original warranty with a period of w repeatedly upon each renewing, warranty issuers could set $w_i = \alpha w_{i-1}$, $\alpha \in (0, 1]$, for $i = 1, 2, \dots$, where w_i is the warranty length for the i -th renewing, and $w_0 = w$. Actually, this defines a new type of renewable warranty, which we refer to as geometric renewable warranty policies. Clearly, a geometric renewable policy is a generalization of a regular renewable policy, which degenerates to the latter when $\alpha = 1$.

The majority of warranties in the market are nonrenewable; for these the warranty cycle, which is the same as the warranty period, is not random, but predetermined (fixed), since the warranty obligation will be terminated as soon as w units of time pass after sale. This type of policies is also known as a fixed-period warranty.

10.2.2 FRW, FRPW, PRW, CMW, and FSW Policies

According to the methods of compensation specified in a warranty contract upon premature failures, there are three basic types of warranties: free replacement warranty (FRW), free repair warranty (FRPW), and pro-rata warranty (PRW). Combination warranty (CMW) contains both features of FRW/FRPW and PRW. Full-service warranty, (FSW), which is also known as preventive maintenance warranty, is a policy that may be offered for expensive deteriorating complex products such as automobiles. Under this type of policies, consumers not only receive free repairs upon premature failures, but also free (preventive) maintenance.

For nonrepairable products, the failed products under warranty will usually be replaced free of charge to consumers. Such a policy is often referred to as a free replacement warranty or an unlimited warranty. In practice, even if a product is technically repairable, sometimes it will be replaced upon failure since repair may not be economically sound. As a result, for inexpensive repairable products, warranty issuers could simply offer FRW policies. Consequently, these inexpensive repairable products can be treated as nonrepairable. However, for repairable products, if the warranty terms specify that, upon a valid warranty claim, the warranty

issuer will repair the failed product to working condition free of charge to buyers, then such a policy is a so-called free repair warranty. In practice, it is not rare that a warranty contract specifies that the warranty issuer would repair or replace a defective product under certain conditions. This is the reason why most researchers do not treat FRW and FRPW separately. Nevertheless, we feel that it is necessary to differentiate these two type of policies based on the following reasoning: first, repair cost is usually much lower than replacement cost except for inexpensive products; secondly, by clearly defining the compensation terms, warranty issuers may establish a better image among consumers, which can surely be helpful for the marketing purpose.

Under a FRW policy, since every failed product within T is replaced by a new one, it is reasonable to model all the subsequent failure times by a single probability distribution. However, under a FRPW, it is necessary to model the repair impact on failure times of a warranted product. If it is assumed that any repair is as-good-as-new (perfect repair), then from the modeling perspective, there is little difference between FRW and FRPW. For deteriorating complex systems, minimal repair is a commonly used assumption. Under this assumption, a repair action restores the system's failure rate to the level at the time epoch when the last failure happened. Minimal repair was first introduced by *Barlow and Proschan* [38]. Changing a broken fan belt on an engine is a good example of minimal repair since the overall failure rate of the car is nearly unchanged. Perfect repair and minimal repair represent two extremes relating to the degree of repair. Realistically, a repair usually makes a system neither as-good-as-new, nor as-bad-as-old (minimal repair), but to a level in between. This type of repair is often referred to as imperfect repair. In the literature of maintenance and reliability, many researchers have studied various maintenance policies considering imperfect repair. A recent review on imperfect maintenance was given by *Pham and Wang* [39]. In the warranty literature, the majority of researchers consider repairs as either perfect or minimal. Little has been done on warranty cost analysis considering imperfect repair.

Both FRW and FRPW policies provide full coverage to consumers in case of product failures within T . In contrast, a PRW policy requires that buyers pay a proportion of the warranty service cost upon a failure within T in exchange for the warranty service such as repair or replacement, cash rebate or a discount on purchasing a new product. The amount that a buyer should pay is usually an increasing function of the product age (duration after the sale). As an example, suppose the average repair/replacement cost per failure is c_s , which could be interpreted as the seller's cost per product without warranty, if a linear pro-rata function is used, then the cost for a buyer upon a failure at time t , $t < w$, is $c_s \frac{t}{w}$. The corresponding warranty cost incurred to the manufacturer is $c_s \left(1 - \frac{t}{w}\right)$. PRW policies are usually

renewable and are offered for relatively inexpensive products such as tires, batteries, and so forth.

Generally speaking, FRW and FRPW policies are in the favor of buyers since manufacturers take all the responsibility of providing products that function properly during the whole warranty cycle [1, p. 221]. In other words, it is the manufacturers that bear all the warranty cost risk. In contrast, for PRW policies manufacturers have the relative advantage with regard to the warranty cost risk. Although they do have to offer cash rebates or discounts to consumers if failures happen during T , they are usually better off no matter what consumers choose to do. If a consumer decides not to file a warranty claim, then the manufacturer saves himself the cash rebate or other type of warranty service. If instead a warranty claim is filed, the manufacturer might enjoy the increase in sales or at least the warranty service cost is shared by the consumer.

To balance the benefits between buyers and sellers, a combination warranty (CMW) that contains both features of FRW/FRPW and PRW policies was created. CMW is a policy that usually includes two warranty periods: a free repair/replacement period w_1 followed by a pro-rata period w_2 . This type of warranties is not rare today because it has significant promotional value to sellers, while at the same time it provides adequate control over the costs for both buyers and sellers [2, p. 12].

For deteriorating complex products, it is essential to perform preventive maintenance to achieve satisfactory reliability performance. Maintenance involves planned and unplanned actions carried out to retain a system at, or restore it to, an acceptable operating condition [40]. Planned maintenance is usually referred to as preventive maintenance, while unplanned maintenance is labeled as corrective maintenance or repair. The burden of maintenance is usually on the consumers, side. In [41], Bai and Pham proposed a renewable full-service warranty for multi-component systems under which the failed component(s) or subsystem(s) will be replaced; in addition, a (preventive) maintenance action will be performed to reduce the chance of future product failures, both free of charge to consumers. They argue that such a policy is desirable for both consumers and manufacturers since consumers receive better warranty service compared to traditional FRPW policies, while at the same time manufacturers may enjoy cost savings due to the improved product reliability by the maintenance actions. By assuming perfect maintenance, they derived the probability distributions and the first two moments of the warranty cost per warranty cycle for series, parallel, series-parallel, and parallel-series systems.

Many researchers have studied warranty-maintenance problems. Among them Chun [42] determined the optimal number of periodic maintenance actions during the warranty period by minimizing the expected warranty cost (EWC). Jack and Daguniar [43] generalized Chun's idea by

considering unequal preventive maintenance intervals. Yeh [31] further extended the work by including the degree of maintenance as one of the decision variables along with the number of maintenance actions and the maintenance schedule. All of these three researches aim to obtain the optimal maintenance warranty to assist manufacturers, decision-making. A related problem is the determination of the optimal maintenance strategy following the expiration of warranty from the consumers, perspective. Dagpunar and Jack [44] studied the problem by assuming minimal repair. Through a general approach, Sahin and Polatoglu [45] discussed both stationary and non-stationary maintenance strategies following the expiration of warranty. They proved the pseudo-convex property of the cost rate function under some mild conditions.

10.2.3 Repair-Limit Warranty

In maintenance literature, many researchers studied maintenance policies set up in such a way that different maintenance actions may take place depending on whether or not some pre-specified limits are met. Three types of limits are usually considered: repair-number-limit, repair-time-limit, and repair-cost-limit. Those maintenance policies are summarized by Wang [36].

Similarly, three types of repair-limit warranties may be considered by manufacturers: repair-number-limit warranty (RNLW), repair-time-limit warranty (RTLW), and repair-cost-limit warranty (RCLW). Under a RNLW, the manufacturer agrees to repair a warranted product up to m times within a period of w . If there are more than m failures within w , the failed product shall be replaced instead of being repaired again. Bai and Pham [46] recently studied the policy under the imperfect-repair assumption. They derived the analytical expressions for the expected value and the variance of warranty cost per product sold through a truncated quasi-renewal-process approach.

AN RTLW policy specifies that, within a warranty cycle T , any failures shall be repaired by the manufacturer, free of charge to consumers. If a warranty service cannot be completed within τ unit of time, then a penalty cost occurs to the manufacturer to compensate the inconvenience of the consumer. This policy was analyzed by Murthy and Asgharizadeh [17] in the context of maintenance service operation.

For a RCLW policy, there is a repair cost limit τ in addition to an ordinary FRPW policy. That is, upon each failure within the warranty cycle T , if the estimated repair cost is greater than τ , then replacement instead of repair shall be provided to the consumer; otherwise, normal repair will be performed. This policy has been studied by Nguyen and Murthy [47] and others.

It should be noted that various repair limits as well as other warranty characteristics such as renewing may be combined together to define a new complex warranty. For example, it is possible to have a renewable repair-time-limit warranty for complex systems. Such combinations define a large set of new warranty policies that may appear in the market in the near future. Further study is needed to explore the statistical behavior of warranty costs of such policies to assist decisions of both manufacturers and consumers.

10.2.4 One-Attribute Warranty and Two-Attribute Warranty

Most warranties in practice are one-attribute, for which the warranty terms are based on product age or product usage, but not both. Compared to one-attribute warranties, two-attribute warranties are more complex since the warranty obligation depends on both the product age and product usage as well as the potential interaction between them. Two-attribute warranties are often seen in automobile industry. For example, Hyundai, the Korean automobile company, is currently offering 10 years/100,000 miles limited FRPW on the powertrain for most of their new models.

One may classify two-attribute warranties according to the shape of warranty coverage region. *Murthy* et al. defined four types of two-attribute warranties labeled as policy A to policy D (Fig. 1 in [48]). The shapes of the warranty regions are rectangular, L-shaped with no limits on age or usage, L-shaped with upper limits on age and usage, and triangular, respectively. Based on the concept of the iso-cost curve, *Chun* and *Tang* [27] proposed a set of two-attribute warranty policies for which the expected present values of future repair costs are the same. Some other plausible warranty regions for two-attribute warranty policies were discussed by *Singpurwalla* and *Wilson* [4].

In general, there are two approaches in the analysis of two-attribute warranties, namely, the one-dimensional (1-D) approach [48] and the two-dimensional (2-D) approach [49]. The 1-D approach assumes a relationship between product age and usage; therefore it eventually converts a two-attribute warranty into a corresponding one-attribute warranty. This approach is used by *Moskowitz* and *Chun* [50], and *Chun* and *Tang* [27]. The 2-D approach does not impose a deterministic relationship between age and usage. Instead, a bivariate probability distribution is employed for the two warranty attributes. *Murthy* et al. [48] followed the idea and derived the expressions for the expected warranty cost per item sold and for the expected life cycle cost based on a two-dimensional renewal processes. *Kim* and *Rao* [51] obtained the analytical expressions for the warranty cost for the policies A and B defined in [48] by considering a bivariate exponential distribution. Perhaps the most comprehensive

study of two-attribute warranties so far is by *Singpurwalla* and *Wilson* [4], in which, through a game-theory set up, they discussed in detail both the optimum price-warranty problem and the warranty reserve determination problem. *Park* and *Pham* [49] recently discussed several warranty cost models for complex systems considering two types of warranty periods such as warranty period and post warranty period subject to minimal repairs. They presented the long run expected cost models per unit time with consideration of both the manufacturer and the customer perspectives and provided the optimum decision variables including warranty period, repair time limit and periodical maintenance cycles that minimizes the total expected cost.

10.3 Evaluation of Warranty Policies

Two phenomena make the study of warranties important. First, warranty has become common practice for manufacturers. According to the survey conducted by *McGuire*, nearly 95% of producers of industrial products provide warranties on all of their product lines [3, p. 1]; secondly, there is a huge amount of money involved in warranty programs. Based on a report by the Society of Mechanical Engineering (www.sme.org), the annual warranty cost is about 6 billion dollars for Ford, General Motors and Chrysler combined in the year 2001.

Among many issues related to warranty, there are two fundamental questions that must be answered, especially for warranty issuers: (1) how much a warranty will cost; (2) how much benefit can be earned from a certain warranty. This section summarizes some ideas and discussions appeared in the literature that are closely related to these two questions.

10.3.1 Warranty Cost Factors

Due to the random nature of many warranty cost factors such as product failure times, warranty cost is also a random variable whose statistical behavior can be determined by establishing mathematical links between warranty factors and warranty cost. There are numerous factors that may be considered in warranty studies. Among them, we believe that the followings are of great importance: the characteristics of warranty policies; warranty service cost per failure; product failure mechanism; impact of warranty service on product reliability; warranty service time; and warranty-claim-related factors.

Different warranty policies may require different mathematical models for warranty cost. One way to model the warranty cost per item sold is through a stochastic counting process $[N(t), t \geq 0]$, which represents the number of failures over time of a warranted product. Let S_1, S_2, \dots be the

subsequent failure times, and denote by C_{S_i} the warranty cost associated with the i -th failure. Assuming that all product failures are claimed, that all claims are valid, and instant warranty service, then the total warranty cost per item sold, $C(w)$, can be expressed as

$$C(w) = \begin{cases} \sum_{i=0}^{N[T(w)]} C_{S_i}, & \text{for } N[T(w)] = 1, 2, \dots \\ 0, & \text{for } N[T(w)] = 0. \end{cases} \quad (10.1)$$

From (10.1), it is clear that the probabilistic behavior of $C(w)$ solely depends on $N[T(w)]$ (the number of failures within a warranty cycle T) and C_{S_i} , as well as the potential interaction between them. In general it is very difficult to determine the distribution of $C(w)$. However, it is possible to obtain the moments of $C(w)$ using modern stochastic process theory and probability theory.

For nonrepairable products or repairable products with a single component, warranty service cost per failure is often assumed to be constant. However, for repairable multi-component products, warranty service cost per failure in general is a random variable whose distribution is related to the product (system) structure and the warranty service cost for each component.

Product (system) failure mechanism can be described by the distributions of subsequent system failure times. This involves the consideration of system structure, the reliability of components, and the impact of repair on components, reliability and system reliability. System structure is essential in determining system reliability. Extensive research on reliability modeling has been done for different systems such as series-parallel systems, parallel-series systems, standby systems, k -out-of- n systems, and so forth, in the literature of reliability [52]. Unfortunately, to our knowledge, there is no complete theory or methodology in warranty that incorporates the consideration of various system structure.

If a warranted product is nonrepairable or the as-good-as-new repair assumption is used for repairable products, then a single failure-time distribution can be adopted to describe the subsequent product failure times under warranty. However, if a warranted product is repairable and repairs are not as-good-as-new, then the failure time distribution(s) of repaired products differ(s) from that of a new product. This situation may be modeled by considering a failure-time distribution for all repaired products different from that of new products [1]. Strictly speaking, distributions of subsequent failure times of a repairable product are distinct, therefore, such an approach can be only viewed as an approximation.

As mentioned in section "Evaluation of Warranty Policies," warranty compensation includes free replacement, free repair or cash rebate. For the case of free replacement, warranty service cost per failure for manufacturers is simply a constant that does not depend on the product failure times. In the case of cash rebate (pro-rata policy), warranty cost

per failure usually relies on product failure time as well as the rebate function. When repair, especially the not as-good-as-new repair, is involved in warranty service, one has to model the repair impact on product reliability, which in turn has a great impact on warranty cost per failure. One way to model subsequent failure times under this situation is to consider them as a stochastic process. Consequently, modern stochastic theory of renewal processes, nonhomogeneous Poisson processes, quasi-renewal processes [40] and general point processes could be applied.

To our knowledge, most warranty literature assumes that warranty service is instant. This may be justified when the warranty service time is small compared to the warranty period or the warranty cycle. A better model is to incorporate explicitly the service times into warranty cost modeling. One recent attempt to include non-zero service time in warranty analysis is by *Murthy and Asgharizadeh* [17]. In this chapter, they developed a game-theoretic formulation to obtain the optimal decision in a maintenance service operation.

Warranty claims-related factors include the response of consumers to product failures and the validation of warranty claims by warranty issuers. It is no secret that not all consumers will make warranty claims even if they are entitled to do so. It is also true that warranty issuers, to serve their own benefits, usually have a formal procedure to validate warranty claims before honoring them. Such situations may be modeled by assigning two new parameters α and β , where α is the probability that a consumer will file a claim upon a failure within T , and β is the proportion of the rejected claims [53].

There are other factors that may be of importance in warranty cost evaluation such as nonconforming product quality [5], multiple modes of failure, censored observations [19], etc. Unfortunately, it is impossible to consider all the factors in one warranty cost model. Even if such a model exists, it would be too complicated to be applied.

10.3.2 Criteria for Comparison of Warranties

Warranty managers usually have several choices among various warranty policies that might be applied to a certain type of products. This requires some basic measures as the criteria to make the comparison among these policies.

There are several measures available, including expected warranty cost (EWC) per product sold, expected discounted warranty cost (EDWC) per warranty cycle, monetary utility function and weighted objective function. EWC and EDWC are more popular than the others since they are easy to understand and can be estimated relatively easily. The key difference between them is that the latter one considers the value of time, an important factor for warranty cost accounting and financial managers.

To our opinion, monetary utility function, $U(x)$, is a better candidate for the purpose of comparing warranty policies. The functional form of $U(x)$ reflects the manufacturer's risk attitude. If a manufacturer is risk-neutral, then $U(x)$ is linear in x . This implies that maximizing $E[U(x)]$ is the same as maximizing $U[E(x)]$. However, manufacturers may be risk-averse if they are concerned about the variations in profit or in warranty cost. For example, a particular manufacturer may prefer a warranty with less cost variation than another with much larger variation in warranty cost if the difference between the EWCs is small. If this is the case, then it can be shown that the corresponding utility function is concave [54]. The main difficulty of the utility theory approach is that utility functions are subjective.

Weighted objective functions could also be used for the purpose of comparing warranties for manufacturers. One commonly used weighted objective function is $E[\pi(x)] - \rho V[\pi(x)]$, where ρ is a positive parameter representing the subjective relative importance of the risk (variance or standard deviation) against the expectation and $\pi(x)$ is the manufacturer's profit for a given warranty policy x . Interestingly, such an objective function coincides to a special case of the utility theory approach when the manufacturer's subjective utility function is assumed to only depend on the first two centered moments of $\pi(x)$ [55, 56].

In the above discussion, the term *warranty cost* refers to the manufacturer's cost per warranted product. In our opinion, this is the fundamental measure for the purpose of evaluating any warranty for manufacturers since it provides precise information on the additional cost incurred to manufacturers due to warranty. An equally useful measure is the discounted warranty cost (DWC) per cycle. This measure incorporates the value of time, therefore it is useful when warranty managers are interested in determining warranty reserve level. It is also of importance to financial managers performing warranty cost analysis.

Some researchers have proposed warranty cost per unit time, or warranty cost rate, as the primary warranty cost measure. As indicated by *Blischke* and *Murthy* [2], warranty cost rate is useful in managing warranty servicing resources, such as parts inventory over time with dynamic sales.

Another related measure is warranty cost over a product life cycle. *Blischke* and *Murthy* named this cost as *life cycle cost-II* (LCC-II) [1]. A product life cycle begins with the launch of the product onto the market and ends when it is withdrawn. For consumers, warranty cost analysis is usually conducted over the life time of a product. In [1], this cost is labeled as *life cycle cost-I* (LCC-I). LCC-I is a consumer-oriented cost measure and it includes elements such as purchase cost, maintenance and repair costs following expiration of a warranty, operating costs as well as disposal costs.

Park and Pham [35] studied the expected warranty cost models for two maintenance policies such as age replacement

and block replacement policies with various warranty policies. For instant, if a failed product is delivered to the warranty service center, the repair service is provided, and if the repair time exceeds the repair time limit the replacement service is provided.

10.3.3 Warranty Cost Evaluation for Complex Systems

Most products (systems), especially expensive ones, are composed of several nonrepairable components. Upon a failure, the common repair practice is to replace failed components instead of replacing the whole system. For such products, warranty may be offered for each of the individual components, or for the whole system. For the former case, the warranty cost modeling and analysis for single-component products can be applied readily. In fact, most warranty literature focuses on the analysis of warranty for single-component systems via a black-box approach. However, for the latter case, it is necessary to investigate warranty with explicit consideration of system structure because evidently system structure has a huge impact on product reliability, therefore it is a crucial factor in warranty cost study. Unfortunately, as indicated by *Chukova* and *Dimitrov* [57, p. 544], so far there has been only limited study on this topic.

Some researchers have discussed the warranty cost modeling for parallel systems. For example, *Ritchken* [58] provided an example of a two-component parallel system under a two-dimensional warranty. *Hussain* and *Murthy* [59] also discussed warranty cost estimation for parallel systems under the setting that uncertain quality of new products may be a concern for the design of warranty programs. *Chukova* and *Dimitrov* [57] presented a two-component parallel system under a FRPW policy. Actually, for nonrepairable parallel systems, the modeling techniques of warranty cost is essentially the same as that of black-box systems unless the system is considered as repairable.

To our knowledge, the only published work about warranty study on series systems is by *Chukova* and *Dimitrov* [57, pp. 579–580]. They derived the EWC per system sold for a two-component series system under a FRPW policy which offers free replacement of the failed component if any system failure happens within the warranty period w . Recently, *Bai and Pham* [41] obtained the first two moments of a renewable FSW policy for series, parallel, series–parallel, and parallel–series systems. The derivation of the first two moments of the DWC of nonrenewable FRPW and PRW policies for minimally repaired series systems can be found in [60].

It is possible to use a Markovian model to analyze warranty cost for complex systems. *Balachandran* et al. [61] dealt with the problem of determining warranty service cost of a three-component system using the Markovian approach.

A similar discussion can be seen in [57] and the references therein. Although this approach is a powerful tool in the literature of reliability, queuing systems, and supply-chain management, there are some limitations in the applications of warranty. First of all, it is difficult to determine the appropriate state space and the corresponding transition matrix for the applications in warranty. Secondly, most Markovian models only provide the analysis of measures in the steady states by assuming infinite horizon. In other words, the statistical behavior of those measures in finite horizon (short-run) is either too difficult to obtain or not of practical interest. However, in warranty study, it is crucial to understand the finite-horizon statistical behavior of warranty cost. Thirdly, warranty claim data as well as reliability data are scarce and costly. Markovian models usually require more data since they contain more parameters than ordinary probability models that could be applied to warranty cost study.

Using a quasi-renewal process [62], Park and Pham [63, 64] discussed several warranty cost models including repairable models with a fixed warranty period for multicomponent systems subject to imperfect repair service. In their models, they considered that: (i) the interoccurrence failure intervals are independent of each other and its follow an exponential distribution; (ii) the failed products are repairable; and (iii) the inspection time that examines whether the failed components need repair services or not is negligible. Using a quasi-renewal process [62, 65], they derived the expected warranty cost and the standard deviation of the warrant cost for systems including parallel-series and series-parallel systems. Sgarbossa and Pham [66] and Pham and Zhang [67] developed several cost models of software systems with considerations of risk and warranty factors to obtain the software testing policies that minimizes the total system costs.

10.3.4 Assessing Warranty Benefits

As mentioned in the introduction, warranty is increasingly used as a promotional device for marketing purposes. Consequently, it is necessary to predict and assess quantitatively the benefit that a manufacturer might generate from a specific warranty [34, p. 189]. For promotional warranties, such benefit is usually realized through the demand side. Manufacturers generally expect that the increase in profit as a result of the increase in sale, which is boosted by warranty, should cover the future warranty cost.

A simple way to quantify the benefit is to model it as a function of the parameter(s) of a warranty policy, for example, w , the warranty period. A linear form and a quadratic form of w were employed by Thomas [34, 68] for this purpose. As he acknowledged, both forms were not well-founded and shared the problem of oversimplification [34, p. 193]. Another approach is to estimate the demand function

empirically. Menezes and Currim [69] posited a general demand function where the quantity sold by a firm offering a warranty with period w is a function of its price, warranty length, advertising, distribution, quality, product feature, and the corresponding values for the firm,s competitor. Based on the data from *Ward,s Automotive Yearbook*, *Consumer Reports*, *Advertising Age*, *Leading National Advertisers*, and other sources during the period 1981–1987, they obtained the price elasticity and the warranty elasticity, which enabled them to obtain the optimal warranty length through maximizing the present value of cumulative future profit over a finite planning horizon. One of the limitations of this approach, as pointed out by the authors, is that it requires the support of historical sales data. As a result, it cannot be applied to new products or existing products without such historical data [69, p. 188].

A related problem of the demand side of warranty is the modeling of sales over time. Mahajan et al. presented several variant diffusion models that may be appropriate for consumer durables [70]. Ja et al. obtained the first two moments of warranty cost in a product life cycle by assuming a nonhomogeneous Poisson sale process [20]. It seems that such models do not allow the interaction between warranty and sales, therefore, they may not be used in estimating warranty benefit.

There is some research (Emons [14], Lutz and Padmanabhan [15], Padmanabhan and Rao [16], etc.) on the demand side of warranty concerning moral hazard, advertising, consumers satisfaction, and so forth. However, compared to the vast warranty literature on estimating total warranty cost, the study on the demand side of warranty is far behind. Hopefully we will see more studies on this aspect in the future.

10.3.5 On the Optimal Warranty Policy

One of the most important objectives of warranty study is to assist warranty management. In particular, in the design phase of a warranty program, there are often a set of warranties that might be appropriate for a specific type of products. The problem faced by warranty managers therefore is how to determine the optimal warranty policy.

An early attempt to address the warranty design problem is based on the concept of life-cycle costing (Blischke [71], Mamer [72]). It is assumed that a consumer requires the product over a certain time period or life cycle from the same producer repeatedly upon each product failure no matter whether under warranty or not. Under this idealized producer–consumer relationship, the producer,s life-cycle profit and the consumer,s life-cycle cost can be calculated. Consequently, a consumer indifference price may be determined by comparing consumer,s life-cycle costs with or without warranty. Similarly, the producer,s indifference price

may be calculated based on the comparison of the life-cycle profits with or without warranty.

An alternative approach is to set up an optimization problem to determine the optimal price and warranty length combination jointly through a game-theoretic perspective. In general, two parties, a warranty issuer and a representative consumer, participate in the game. The latter acts as a follower who responds rationally to each potential warranty offer by maximizing his/her utility. The former, as a leader, makes the decision on the optimal warranty strategy, which maximizes the expected profit, based on the anticipated rational response by the consumer. *Singpurwalla and Wilson* [4] studied two-attribute warranties through this approach. Some others references are *Chun and Tang* [73], *DeCroix* [12], *Glickman and Berger* [74], *Ritchken* [75], *Thomas* [68], and the references therein. In the context of production planning and marketing, *Mitra and Patankar* [76] presented a multi-criteria model that could be used in warranty design.

Now, we present a general formulation of the warranty design problem with some discussion, which may raise more interest among researchers and practitioners for further study.

Let $\Psi = \{\psi_1, \psi_2, \dots, \psi_n\}$ represent the set of appropriate warranty policies for a given type of products. Policy ψ_i may contain more than one parameter. Denote by w_i the set of warranty parameters for ψ_i ; then we can represent ψ_i by $\psi(w_i)$ or w_i . If w_i contains only one parameter, say, w_i , the warranty period, then $w_i = \{w_i\}$. Denote by $p(w_i)$ the selling price under the policy ψ_i , and by $C_j(w_i)$ the random warranty cost for the j -th product sold under the policy ψ_i . Let p_0 be the production cost per unit (not including the warranty cost), then the optimal warranty policy $\psi(w^*)$ may be obtained by solving

$$\begin{aligned} & \max_{\{w_i, \forall i=1,2,\dots,n\}} E \{U[\pi(w_i)]\} \\ & \text{s.t. } w_i^l \leq w_i \leq w_i^u, \forall i, i = 1, 2, \dots, n \\ & P \left[\sum_{j=1}^{d(w_i)} C_j(w_i) \geq R_0 \right] \leq \alpha, \forall i, i = 1, 2, \dots, n, \end{aligned}$$

where

$U(\cdot)$ is the monetary utility function that reflects the risk attitude of the manufacturer. It is a linear function if the manufacturer is risk-neutral and a concave function in the case of a risk-averse manufacturer;

$$\pi(w_i) = \sum_{j=1}^{d(w_i)} [p(w_i) - p_0 - C_j(w_i)],$$

w_i^l, w_i^u : are some lower and upper bounds of w_i ;
 $d(w_i)$ represents the demand function for $\psi(w_i)$;
 R_0 is the predetermined warranty budget level; and
 α is the risk-tolerance level of the manufacturer with regard to R_0 .

One should note that the second set of constraints is actually related to *value at risk* (VaR), a concept widely used in risk management, which indicates the maximum percentage value of an asset that could be lost during a fixed period within a certain confidence level [77]. It is reasonable to assume that manufacturers want to control VaR such that the probability that the total warranty cost is over the budget is within the accepted level α .

Solving the optimization problem might be a challenge. First of all, it is difficult to determine the demand function $d(w_i)$, although it is possible to estimate it through marketing surveys or historical data. Secondly, it is required that warranty managers have complete knowledge of the selling price $p(w_i)$. This requires a pricing strategy in the design phase of warranty. It should be noted that we could have considered $p(w_i)$ as one of the decision variables, but this makes the problem more complicated. Besides, it is not rare in practice that the price is simply set by adding a fixed margin over the estimated production cost with warranty. Thirdly, it is required that the probability distribution of warranty cost should be known. Little research has been done with regard to this issue except *Polatoglu and Sahin* [26] and *Sahin and Polatoglu* [78]. In general, numerical methods are required for this purpose. Fourthly, the problem is formulated as a nonlinear optimization problem with some constraints, which may be solved by nonlinear optimization software such as GAMS. However, in general there is no guarantee of the existence of a global optimal solution.

10.4 Concluding Remarks

A warranty problem, by its nature, is a multi-disciplinary research topic. Many researchers ranging from the industry engineer, economist, statistician, to marketing researchers have contributed greatly to warranty literature. In this chapter, we present an overview of warranty policies, focusing on the cost and benefit analysis from warranty issuers, perspective. Although we have successfully addressed several problems in this area, there are still a lot of opportunities for future research, a few of which are listed below:

- To advance warranty optimization models and perform empirical study based on the new developed models.
- To develop and apply efficient algorithms to solve warranty optimization problems.
- To propose and analyze new warranty policies appropriate for complex systems.
- To Study the distribution and the moments of discounted warranty cost for various policies.
- Warranty cost modeling for systems with more complex structures, including standby systems, bridge systems, network systems, etc.

- Develop warranty models considering failure dependency between components due to environmental impact.

References

- Blischke, W.R., Murthy, D.N.P.: *Warranty Cost Analysis*. Marcel Dekker, New York (1994)
- Blischke, W.R., Murthy, D.N.P. (eds.): *Product Warranty Handbook*. Marcel Dekker, New York (1996)
- McGuire, E.P.: *Industrial Product Warranties: Policies and Practices*. The Conference Board, New York (1980)
- Singpurwalla, N.D., Wilson, S.: The warranty problem: its statistical, game theoretic aspects. *SIAM Rev.* **35**, 17–42 (1993)
- Chen, J., Yao, D.D., Zheng, S.: Quality control for products supplied with warranty. *Oper. Res.* **46**, 107–115 (1988)
- Djameludin, I., Murthy, D.N.P.: Quality control through lot sizing for items sold with warranty. *Int. J. Prod. Econ.* **33**, 97–107 (1994)
- Hegde, G.G., Kubat, P.: Diagnostic design: a product support strategy. *Eur. J. Oper. Res.* **38**, 35–43 (1989)
- Mi, J.: Warranty, burn-in. *Nav. Res. Logist.* **44**, 199–210 (1996)
- Murthy, D.N.P., Hussain, A.Z.M.O.: Warranty, optimal redundancy design. *Eng. Optim.* **23**, 301–314 (1993)
- Nguyen, D.G., Murthy, D.N.P.: Optimal reliability allocation for products sold under warranty. *Eng. Optim.* **13**, 35–45 (1988)
- Sahin, I.: Conformance quality, replacement costs under warranty. *Prod. Oper. Manag.* **2**, 242–261 (1993)
- DeCroix, G.A.: Optimal warranties, reliabilities, prices for durable goods in an oligopoly. *Eur. J. Oper. Res.* **112**, 554–569 (1999)
- Emons, W.: Warranties, moral hazard, the lemons problem. *J. Econ. Theory.* **46**, 16–33 (1988)
- Emons, W.: On the limitation of warranty duration. *J. Ind. Econ.* **37**, 287–301 (1989)
- Lutz, M.A., Padmanabhan, V.: Warranties, extended warranties, product quality. *Int. J. Ind. Organ.* **16**, 463–493 (1998)
- Padmanabhan, V., Rao, R.C.: Warranty policy, extended service contracts: theory, an application to automobiles. *Mark. Sci.* **12**, 97–117 (1993)
- Murthy, D.N.P., Asgharizadeh, E.: Optimal decision making in a maintenance service operation. *Eur. J. Oper. Res.* **116**, 259–273 (1999)
- Frees, E.W.: Warranty analysis, renewal function estimation. *Nav. Res. Logist. Quart.* **33**, 361–372 (1986)
- Frees, E.W.: Estimating the cost of a warranty. *J. Bus. Econ. Stat.* **6**, 79–86 (1988)
- Ja, S., Kulkarni, V., Mitra, A., Patankar, G.: Warranty reserves for non-stationary sales processes. *Nav. Res. Logist.* **49**, 499–513 (2002)
- Kalbfleisch, J.D., Lawless, J.F., Robinson, J.A.: Methods for the analysis, prediction of warranty claims. *Technometrics.* **33**, 273–285 (1991)
- Kao, E.P.C., Smith, M.S.: Computational approximations of renewal process relating to a warranty problem: the case of phase-type lifetimes. *Eur. J. Oper. Res.* **90**, 156–170 (1996)
- Kao, E.P.C., Smith, M.S.: On excess, current, total life distributions of phase-type renewal processes. *Nav. Res. Logist.* **39**, 789–799 (1992)
- Menzefricke, U.: On the variance of total warranty claims. *Comm. Stat. Theory Methods.* **21**, 779–790 (1992)
- Patankar, J.G., Worm, G.H.: Prediction intervals for warranty reserves, cash flows. *Manag. Sci.* **27**, 237–241 (1981)
- Polatoglu, H., Sahin, I.: Probability distribution of cost, revenue, profit over a warranty cycle. *Eur. J. Oper. Res.* **108**, 170–183 (1998)
- Chun, Y.H., Tang, K.: Cost analysis of two-attribute warranty policies based on the product usage rate. *IEEE Trans. Eng. Manag.* **46**, 201–209 (1999)
- Lam, Y., Lam, P.K.W.: An extended warranty policy with options open to consumers. *Eur. J. Oper. Res.* **131**, 514–529 (2001)
- Wang, C., Sheu, S.: Optimal lot sizing for products sold under free-repair warranty. *Eur. J. Oper. Res.* **164**, 367–377 (2005)
- Yeh, R.H., Ho, W.T., Tseng, S.T.: Optimal production run length for products sold with warranty. *Eur. J. Oper. Res.* **120**, 575–582 (2000)
- Yeh, R.H., Lo, H.: Optimal preventive-maintenance warranty policy for repairable products. *Eur. J. Oper. Res.* **134**, 59–69 (2001)
- Murthy, D.N.P.: Product warranty management—III: a review of mathematical models. *Eur. J. Oper. Res.* **62**, 1–34 (1992)
- Murthy, D.N.P., Djameludin, I.: New product warranty: a literature review. *Int. J. Prod. Econ.* **79**, 231–260 (2002)
- Thomas, M.U., Rao, S.S.: Warranty economic decision models: a summary, some suggested directions for future research. *Oper. Res.* **47**, 807–820 (1999)
- Park, M., Pham, H.: Cost models for age replacement policies and block replacement policies under warranty. *Appl. Math. Model.* **40**(9), 5689–5702 (2016)
- Wang, H., Pham, H.: *Reliability and Optimal Maintenance*. Springer, London (2006)
- Blischke, W.R., Murthy, D.N.P.: Product warranty management—I: a taxonomy for warranty policies. *Eur. J. Oper. Res.* **62**, 127–148 (1993)
- Barlow, R.E., Proschan, F.: *Mathematical Theory of Reliability*. Wiley, New York (1965)
- Pham, H., Wang, H.: Imperfect maintenance. *Eur. J. Oper. Res.* **94**, 425–438 (1996)
- H. Wang: Reliability and maintenance modeling for systems with imperfect maintenance and dependence. Ph.D. Thesis (Rutgers University, Piscataway, 1997) (unpublished)
- Bai, J., Pham, H.: Cost analysis on renewable full-service warranties for multi-component systems. *Eur. J. Oper. Res.* **168**, 492–508 (2006)
- Chun, Y.H.: Optimal number of periodic preventive maintenance operations under warranty. *Reliab. Eng. Syst. Saf.* **37**, 223–225 (1992)
- Jack, N., Dagpunar, J.S.: An optimal imperfect maintenance policy over a warranty period. *Microelectron. Reliab.* **34**, 529–534 (1994)
- Dagpunar, J.S., Jack, N.: Optimal repair-cost limit for a consumer following expiry of a warranty. *IMA J. Math. Appl. Bus. Ind.* **4**, 155–161 (1992)
- Sahin, I., Polatoglu, H.: Maintenance strategies following the expiration of warranty. *IEEE Trans. Reliab.* **45**, 221–228 (1996)
- Bai, J., Pham, H.: Repair-limit risk-free warranty policies with imperfect repair. *IEEE Trans. Syst. Man Cybern. A.* **35**(6), 765–772 (2005)
- Nguyen, D.G., Murthy, D.N.P.: Optimal replacement-repair strategy for servicing products sold under warranty. *Eur. J. Oper. Res.* **39**, 206–212 (1989)
- Murthy, D.N.P., Iskandar, B.P., Wilson, R.J.: Two dimensional failure-free warranty policies: two-dimensional point process models. *Oper. Res.* **43**, 356–366 (1995)
- Park, M., Pham, H.: Warranty cost analysis for k-out-of-n systems with 2-D warranty. *IEEE Trans. Syst. Man Cybern. A.* **42**(4), 947–957 (2012)
- Moskowitz, H., Chun, Y.H.: A Poisson regression model for two-attribute warranty policies. *Nav. Res. Logist.* **41**, 355–376 (1994)
- Kim, H.G., Rao, B.M.: Expected warranty cost of two-attribute free-replacement warranties based on a bivariate exponential distribution. *Comput. Ind. Eng.* **38**, 425–434 (2000)
- Pham, H.: *System Software Reliability*. Springer, London (2006)

53. Lee Hill, V., Beall, C.W., Blischke, W.R.: A simulation model for warranty analysis. *Int. J. Prod. Econ.* **16**, 463–491 (1998)
54. Kreps, D.M.: *A Course in Microeconomic Theory*. Princeton University Press, Princeton (1990)
55. Markowitz, H.: *Portfolio Selection*. Yale University Press, Yale (1959)
56. Ritchken, P.H., Tapiero, C.S.: Warranty design under buyer, seller risk aversion. *Nav. Res. Logist. Quart.* **33**, 657–671 (1986)
57. Chukova, S., Dimitrov, B.: Warranty analysis for complex systems. In: Blischke, W.R., Murthy, D.N.P. (eds.) *Product Warranty Handbook*, pp. 543–584. Marcel Dekker, New York (1996)
58. Ritchken, P.H.: Optimal replacement policies for irreparable warranted item. *IEEE Trans. Reliab.* **35**, 621–624 (1986)
59. Hussain, A.Z.M.O., Murthy, D.N.P.: Warranty, redundancy design with uncertain quality. *IEEE Trans.* **30**, 1191–1199 (1998)
60. Bai, J., Pham, H.: Discounted warranty cost of minimally repaired series systems. *IEEE Trans. Reliab.* **53**(1), 37–42 (2004)
61. Balachandran, K.R., Maschmeyer, R.A., Livingstone, J.L.: Product warranty period: a Markovian approach to estimation, analysis of repair, replacement costs. *Account. Rev.* **1**, 115–124 (1981)
62. Wang, H., Pham, H.: A quasi renewal process and its applications in imperfect maintenance. *Int. J. Syst. Sci.* **27**(10), 1055–1062 (1996)
63. Park, M., Pham, H.: Warranty cost analyses using quasi-renewal processes for multi-component systems. *IEEE Trans. Syst. Man Cybern. A.* **40**(6), 1329–1340 (2010)
64. Park, M., Pham, H.: Altered quasi-renewal concepts for modeling renewable warranty costs with imperfect repairs. *Math. Comput. Model.* **52**(2010), 1435–1450 (2010)
65. Park, M., Pham, H.: A new warranty policy with failure times and warranty servicing times. *IEEE Trans. Reliab.* **61**(3), 822–831 (2012)
66. Sgarbossa, F., Pham, H.: A cost analysis of systems subject to random field environments and reliability. *IEEE Trans. Syst. Man Cybern. C.* **40**(4), 429–437 (2010)
67. Pham, H., Zhang, X.: A software cost model with warranty and risk costs. *IEEE Trans. Comput.* **48**(1), 71–75 (1999)
68. Thomas, M.U.: Optimum warranty policies for nonreparable items. *IEEE Trans. Reliab.* **32**, 283–288 (1983)
69. Menezes, M.: An approach for determination of warranty length. *Int. J. Res. Mark.* **9**, 177–195 (1992)
70. Mahajan, V., Muller, E., Wind, Y.: *New-Product Diffusion Models*. Kluwer Academic, Dordrecht (2000)
71. Blischke, W.R., Scheuer, E.M.: Calculating the cost of warranty policies as a function of estimated life distributions. *Nav. Res. Logist. Quart.* **28**, 193–205 (1975)
72. Mamer, J.W.: Discounted, per unit costs of product warranty. *Manag. Sci.* **33**, 916–930 (1987)
73. Chun, Y.H., Tang, K.: Determining the optimal warranty price based on the producer's, customers, risk preferences. *Eur. J. Oper. Res.* **85**, 97–110 (1995)
74. Glickman, T.S., Berger, P.D.: Optimal price, protection period decisions for a product under warranty. *Manag. Sci.* **22**, 1381–1390 (1976)
75. Ritchken, P.H.: Warranty policies for non-repairable items under risk aversion. *IEEE Trans. Reliab.* **34**, 147–150 (1985)
76. Mitra, A., Patankar, J.G.: An integrated multicriteria model for warranty cost estimation, production. *IEEE Trans. Eng. Manag.* **40**, 300–311 (1993)
77. Jorion, P.: *Value-at-Risk: the New Benchmark for Managing Financial Risk*. McGraw-Hill, New York (2000)
78. Sahin, I., Polatoglu, H.: Distributions of manufacturer's, user's replacement costs under warranty. *Nav. Res. Logist.* **42**, 1233–1250 (1995)



Dr. Hoang Pham is a Distinguished Professor and former Chairman of the Department of Industrial and Systems Engineering at Rutgers University, New Jersey, USA. He is the author or coauthor of 7 books and has published over 200 journal articles, 100 conference papers, and edited 20 books. His numerous awards include the 2009 IEEE Reliability Society *Engineer of the Year Award*. He is a Fellow of the IEEE and IISE.



Dr. Jun Bai is a VP at Barclays Bank US. He is an analytical leader of extensive financial industry experience in Stress Testing and CCAR, risk strategy development, marketing analytics and statistical modeling. He obtained his Ph.D. in industrial and systems engineering from Rutgers – the State University of New Jersey. Currently his research activities focus on stress testing, risk management, and statistical modelling in the financial industry.

Part II

Process Monitoring and Improvement



Statistical Methods for Quality and Productivity Improvement

11

Wei Jiang, Terrence E. Murphy, Kwok-Leung Tsui, and Yang Zhao

Contents

11.1 Statistical Process Control for Single Characteristics	200
11.1.1 SPC for I.I.d. Processes.....	201
11.1.2 SPC for Autocorrelated Processes.....	201
11.1.3 SPC Versus APC.....	203
11.1.4 SPC for Automatically Controlled Processes.....	204
11.1.5 Design of SPC Methods: Efficiency Versus Robustness.....	205
11.1.6 SPC for Multivariate Characteristics.....	205
11.1.7 SPC for Profile Monitoring.....	207
11.2 Design of Experiment and Robust Parameter Design	207
11.2.1 Robust Design for Single Responses.....	207
11.2.2 Robust Design for Multiple Responses.....	211
11.2.3 Dynamic Robust Design.....	212
11.2.4 Applications of Robust Design.....	213
11.3 Reliability and Prognostics and Health Management	213
11.3.1 Prognostics and Health Management.....	213
11.3.2 Systems Health Monitoring and Management.....	214
References	216

Abstract

The first section of this chapter introduces statistical process control (SPC) and robust design (RD), two important statistical methodologies for quality and productivity improvement. Section 11.1 describes in-depth SPC theory and tools for monitoring independent and autocorrelated data with a single quality characteristic. The relationship between SPC methods and automatic process control methods is discussed and differences in their philosophies, techniques, efficiencies, and design are contrasted. SPC methods for monitoring multivariate quality characteristics are also briefly reviewed.

Section 11.2 considers univariate RD, with emphasis on experimental design, performance measures and modeling of the latter. Combined and product arrays are featured and performance measures examined, include signal-to-noise ratios SNR, PerMIAs, process response, process variance, and desirability functions. Of central importance is the decomposition of the expected value of squared-error loss into variance and off-target components which sometimes allows the dimensionality of the optimization problem to be reduced. Besides, this section deals with multivariate RD and demonstrates that the objective function for the multiple characteristic case is typically formed by additive or multiplicative combination of the univariate objective functions, and lists RD case studies originating from applications in manufacturing, reliability, and tolerance design.

Section 11.3 discusses the mainstream methods used in the prognostics and health management (PHM) framework, including updated research from the literatures of both statistical science and engineering. Additionally, this section provides an overview of the systems health monitoring and management (SHMM) framework, discusses its basic structure, and lists several applications of SHMM to complex systems and to critical components within the context of a big data environment.

W. Jiang
Antai College of Economics and Management, Shanghai Jiao Tong University, Shanghai, China
e-mail: jiangwei@sjtu.edu.cn

T. E. Murphy
Biostatistics and Bioinformatics Department of Public Health Sciences, Pennsylvania State University College of Medicine, Hershey, PA, USA
e-mail: tmurphy11@pennstatehealth.psu.edu

K.-L. Tsui (✉)
Grado Department of Industrial and Systems Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA
e-mail: klttsui@vt.edu

Y. Zhao
School of Public Health (Shenzhen), Sun Yat-sen University, Shenzhen, China
e-mail: zhaoy393@mail.sysu.edu.cn

Keywords

Control chart · Minimum mean square error · Robust design · Desirability function · Statistical process control · Health monitoring and management

In the current international marketplace, continuous quality improvement is pivotal for maintaining a competitive advantage. Although quality improvement activities are most efficient and cost-effective when implemented as part of the design and development stages (off-line), on-line activities such as statistical process control (SPC) are vital for maintaining quality during manufacturing.

Statistical process control (SPC) is an effective tool for achieving process stability and improving process capability through variation reduction. Primarily, SPC is used to classify sources of process variation as either common cause or assignable cause. Common cause variations are inherent to a process and can be described implicitly or explicitly by stochastic models. Assignable cause variations are unexpected and difficult to predict beforehand. The basic idea of SPC is to quickly detect and correct assignable cause variation before quality deteriorates and defective units are produced. The primary SPC tool was developed in the 1920s by Walter Shewhart of Bell Telephone Laboratories and has been tremendously successful in manufacturing applications [1–3].

Robust design (RD) is a systematic methodology that uses statistical experimental design to improve the design of products and processes. By making product and process performance insensitive (robust) to hard-to-control disturbances (noise), robust design simultaneously improves product quality, the manufacturing process, and reliability. The RD method was originally developed by the Japanese quality consultant, *Genichi Taguchi* [4]. Taguchi's 1980 introduction of robust parameter design to several major American industries resulted in significant quality improvements in product and process design [5]. Since then, a great deal of research on RD has improved related statistical techniques and clarified underlying principles.

In addition, many RD case studies have demonstrated phenomenal cost savings. In the electronics industry, *Kackar* and *Shoemaker* [6] reported a 60% process variance reduction; *Phadke* [5] reported a fourfold reduction in process variance and a twofold reduction in processing time – both from running simple RD experiments. In other industries, the American Supplier Institute (1983–1990) reported a large number of successful case studies in robust design.

Although most data is multivariate in nature, research in both areas has largely focused on normally distributed univariate characteristics (responses). *Montgomery* and *Woodall* [2] present a comprehensive panel discussion on SPC (see also *Woodall* and *Montgomery* [7]) and multivariate methods are reviewed by *Lowry* and *Montgomery* [8] and *Mason* [9].

Seminal research papers on RD include *Kackar* [10], *Leon* et al. [11], *Box* [12], *Nair* [13], and *Tsui* [14]. RD problems with multiple characteristics are investigated by *Logothetis* and *Haigh* [15], *Pignatiello* [16], *Elsayed* and *Chen* [17], and *Tsui* [18]. This research has yielded techniques allowing engineers to effectively implement SPC and RD in a host of applications.

This chapter briefly revisits the major developments in both SPC and RD that have occurred over the last 30 years and suggests future research directions while highlighting multivariate approaches. Section 11.1 covers SPC of univariate and multivariate random variables for both Shewhart (including \bar{x} and s charts) and non-Shewhart approaches (CUSUM and EWMA), while assessing the effects of autocorrelation and automatic process control. Section 11.2 considers univariate RD, emphasizing performance measures and modeling for loss functions, dual responses, and desirability functions; deals with multivariate and dynamic RD; and recaps RD case studies from the statistics literature in manufacturing, process control, and tolerance design. Finally, Sect. 11.3 provides an overview of PHM and SHMM framework.

11.1 Statistical Process Control for Single Characteristics

The basic idea in statistical process control is a binary view of the state of a process; in other words, it is either running satisfactorily or not. *Shewhart* [19] asserted that the process state is related the type of variation manifesting itself in the process. There are two types of variation, called common cause and assignable or special cause variation. Common cause variation refers to the assumption that “future behavior can be predicted within probability limits determined by the common cause system” [20]. Special cause variation refers to “something special, not part of the system of common causes” [21]. A process that is subject only to common cause variation is “statistically” in control, since the variation is inherent to the process and therefore eliminated only with great difficulty. The objective of statistical process control is to identify and remove special cause variation as quickly as possible.

SPC charts essentially mimic a sequential hypothesis test to distinguish assignable cause variation from common cause variation. For example, a basic mathematical model behind SPC methods for detecting change in the mean is

$$X_t = \eta_t + Y_t,$$

where X_t is the measurement of the process variable at time t , and η_t is the process mean at that time. Here Y_t represents variation from the common cause system. In some

applications, Y_t can be treated as an independently and identically distributed (iid) process. With few exceptions, the mean of the process is constant except for abrupt changes, so

$$\eta_t = \eta + \mu_t,$$

where η is the mean target and μ_t is zero for $t < t_0$ and has nonzero values for $t \geq t_0$. For analytical simplicity step changes are often assumed; in other words μ_t remains at a new constant level μ for $t \geq t_0$.

11.1.1 SPC for I.I.d. Processes

The statistical goal of SPC control charts is to detect the change point t_0 as quickly as possible and trigger corrective action to bring the process back to the quality target. Among many others, the Shewhart chart, the EWMA chart, and the CUSUM chart are three important and widely used control charts.

Shewhart Chart

The Shewhart control chart monitors the process observations directly,

$$W_t = X_t - \eta.$$

Assuming that the standard deviation of W_t is σ_W , the stopping rule of the Shewhart chart is defined as $|W_t| > L\sigma_W$, where L is prespecified to maintain particular probability properties.

EWMA Chart

Roberts [22] introduces a control charting algorithm based on the exponentially weighted moving average of the observations,

$$W_t = \sum_{i=0}^{\infty} w_i (X_{t-i} - \eta),$$

where $w_i = \lambda(1 - \lambda)^i$, ($0 < \lambda \leq 1$). It can be rewritten as

$$W_t = (1 - \lambda)W_{t-1} + \lambda(X_t - \eta), \quad (11.1)$$

where $W_0 = 0$ or the process mean. The stopping rule of the EWMA chart is $|W_t| > L\sigma_W$ where $\sigma_W = \sqrt{\lambda/(2 - \lambda)}\sigma_X$. The Shewhart chart is a special case of the EWMA chart with $\lambda = 1$. When the underlying process is i.i.d., the EWMA chart with small λ values is sensitive to the detection of small and medium shifts in mean [23].

CUSUM Chart

Page [24] introduces the CUSUM chart as a sequential probability test. It can be simply obtained by letting λ approach zero in (11.1). The CUSUM algorithm assigns equal weights to past observations, and its tabular form consists of two quantities,

$$\begin{aligned} W_t^+ &= \max[0, W_{t-1}^+ + (X_t - \eta) - k\sigma_X], \\ W_t^- &= \min[0, W_{t-1}^- + (X_t - \eta) - k\sigma_X], \end{aligned}$$

where $W_0^+ = W_0^- = 0$. It can be shown that the CUSUM chart with $k = \mu/2$ is optimal for detecting a mean change in μ when the observations are i.i.d.

Because of the randomness of the observations, these control charts may trigger false alarms – out-of-control signals issued when the process is still in control. The expected number of units measured between two successive false alarms is called the *in-control* average run length $(ARL)_0$. When a special cause presents itself, the expected period before a signal is triggered is called the *out-of-control* average run length $(ARL)_1$. The ideal control chart has a long ARL_0 and a short ARL_1 . The Shewhart chart typically uses the constant $L = 3$, so that the in-control ARL is 370 when the underlying process is i.i.d. with normal distribution.

These SPC charts are very effective for monitoring the process mean when the process data is i.i.d. It has been shown that the Shewhart chart is sensitive for detecting large shifts while the EWMA and CUSUM charts are sensitive to small shifts [23]. However, a fundamental assumption behind these SPC charts is that the common cause variation is free of serial correlation. Due to the prevalence of advanced sensing and measurement technology in manufacturing processes, the assumption of independence is often invalid. For example, measuring critical in-process dimensions is now possible on every unit in the production of discrete parts. In continuous process production systems, the presence of inertial elements such as tanks, reactors, and recycle streams often result in significant serial correlation in the measured variables. Serial correlation creates many challenges and opportunities for SPC methodologies.

11.1.2 SPC for Autocorrelated Processes

Traditional SPC charts have been shown to function poorly while monitoring and controlling serially correlated processes [25, 26]. To accommodate autocorrelation, the following time series methods have been proposed.

Modifications of Traditional Methods

One common SPC strategy is to plot the autocorrelated data on traditional charts whose limits have been modified to account for the correlation. Johnson and Bagshaw [27] and

Bagshaw and Johnson [28] consider the effects of autocorrelation on CUSUM charts using the weak convergence of cumulative sums to a Wiener process. Another alternative is the exponentially weighted moving average chart for stationary processes (EWMAST) studied by Zhang [29]. Jiang et al. [30] extend this to a general class of control charts based on autoregressive moving average (ARMA) charts. The monitoring statistic of an ARMA chart is defined to be the result of a *generalized* ARMA(1, 1) process applied to the underlying process $\{X_t\}$,

$$\begin{aligned} W_t &= \theta_0 X_t - \theta X_{t-1} + \phi W_{t-1} \\ &= \theta_0 (X_t - \beta X_{t-1}) + \phi W_{t-1}, \end{aligned} \quad (11.2)$$

where $\beta = \theta/\theta_0$ and θ_0 is chosen so that the sum of the coefficients is unity when W_t is expressed in terms of the X_t 's, so $\theta_0 = 1 + \theta - \phi$. The authors show that these charts exhibit good performance when the chart parameters are chosen appropriately.

Forecast-Based Monitoring Methods

Forecast-based charts started with the special-cause charts (SCC) proposed by Alwan and Roberts [31]. The general idea is to first apply a one-step-ahead predictor to the observation $\{X_t\}$ and then monitor the corresponding prediction error,

$$W_t = e_t, \quad (11.3)$$

where $e_t = X_t - \hat{X}_t$ is the forecast error of predictor \hat{X}_t . The SCC method is the first example that uses minimum mean squared error (MMSE) predictors and monitors the MMSE residuals. When the model is accurate, the MMSE prediction errors are approximately uncorrelated. This removal of correlation means that control limits for the SCC can be easily calculated from traditional Shewhart charts, EWMA charts, and CUSUM charts. Another advantage of the SCC method is that its performance can be analytically approximated.

The SCC method has attracted considerable attention and has been extended by many authors. Among them, Harris and Ross [25] and Superville and Adams [32] investigate process monitoring based on the MMSE prediction errors for simple autoregressive [AR(1)] models; Wardell et al. [33, 34] discuss the performance of SCC for ARMA(1, 1) models; and Vander Wiel [35] studies the performance of SCC for integrated moving average [IMA(0, 1, 1)] models. SCC methods perform poorly when detecting small shifts since a constant mean shift always results in a dynamic shift pattern in the error term.

In general this approach can be applied to any predictor. Montgomery and Mastrangelo [36] recommend the use of EWMA predictors in the SCC method (hereafter called the M–M chart). Jiang et al. [37] propose the use of

proportional-integral-derivative (PID) predictors

$$\begin{aligned} \hat{X}_t &= \hat{X}_{t-1} + (k_p + k_i + k_d)e_{t-1} \\ &\quad - (k_p + 2k_d)e_{t-2} + k_d e_{t-3}, \end{aligned} \quad (11.4)$$

where k_p , k_i , and k_d are parameters of the PID controller defined in Sect. 11.1.3. The family of PID-based charts includes the SCC, EWMA, and M–M charts as special cases. Jiang et al. [37] show that the predictors of the EWMA chart and M–M chart may sometimes be inefficient and the SCC over-sensitive to model deviation. They also show that the performance of the PID-based chart is affected by the choice of chart parameters. For any given underlying process, one can therefore tune the parameters of the PID-based chart to optimize its performance.

GLRT-Based Multivariate Methods

Since forecast-based residual methods monitor a single statistic e_t , they often suffer from the problem of a narrow “window of opportunity” when the underlying process is positively correlated [35]. If the shift occurrence time is known, the problem can be alleviated by including more historical observations/residuals in the test. This idea was first proposed by Vander Wiel [35] using a generalized likelihood ratio test (GLRT) procedure. Assuming residual signatures $\{\delta_i\}$ when a shift occurs, the GLRT procedure based on residuals is

$$W_t = \max_{0 \leq k \leq p-1} \left| \sum_{i=0}^k \delta_i e_{t-k+i} \right| / \sqrt{\sum_{i=0}^k \delta_i^2}, \quad (11.5)$$

where p is the prespecified size of the test window. Apley and Shi [38] show that this procedure is very efficient in detecting mean shifts when p is sufficiently large. Similar to the SCC methods, this is model-based and the accuracy of signature strongly depends on the window length p . If p is too small and a shift is not detected within the test window, the signature in (11.5) might no longer be valid and the test statistic no longer efficient.

Note that a step mean shift at time $t - k + 1$ results in a signature

$$d_k = \left(0, \dots, 0, \overbrace{1, \dots, 1}^k \right)' \quad (1 \leq k \leq p)$$

and

$$d_k = (1, 1, \dots, 1)' \quad (k > p)$$

on $U_t = (X_{t-p+1}, X_{t-p+2}, \dots, X_t)'$. To test these signatures, the GLRT procedure based on observation vector W_t

is defined as

$$W_t = \max_{0 \leq k \leq p-1} |d'_k \Sigma_U^{-1} U_t| / \sqrt{d'_k \Sigma_U^{-1} d_k}, \quad (11.6)$$

where Σ_U is the covariance matrix of U_t . Jiang [39] points out that this GLRT procedure is essentially model-free and always matches the true signature of U_t regardless of the timing of the change point. If a non-step shift in the mean occurs, multivariate charts such as Hotelling's T^2 charts can be developed accordingly [40].

Monitoring Batch Means

One of the difficulties with monitoring autocorrelated data is accounting for the underlying autocorrelation. In simulation studies, it is well known that batch means reduce autocorrelation within data. Motivated by this idea, Runger and Willemain [41, 42] use a weighted batch mean (WBM) and a unified batch mean (UBM) to monitor autocorrelated data. The WBM method weighs the mean of observations, defines batch size so that autocorrelation among batches is reduced to zero and requires knowledge of the underlying process model [43]. The UBM method determines batch size so that autocorrelation remains below a certain level and is "model-free". Runger and Willemain show that the UBM method is simple and often more cost-effective in practice.

Batch-means methods not only develop statistics based on batch-means, but also provide variance estimation of these statistics for some commonly used SPC charts. Alexopoulos et al. [44] discuss promising methods for dealing with correlated observations including nonoverlapping batch means (NBM), overlapping batch means (OBM) and standardized time series (STS).

11.1.3 SPC Versus APC

Automatic process control (APC) complements SPC as a variation reduction tool for manufacturing industries. While SPC techniques are used to reduce unexpected process variation by detecting and removing the cause of variation, APC techniques are used to reduce systematic variation by employing feedforward and feedback control schemes. The relationships between SPC and APC are important to both control engineers and quality engineers.

Feedback Control Versus Prediction

The feedback control scheme is a popular APC strategy that uses the deviation of output from target (set-point) to signal a disturbance of the process. This deviation or error is then used to compensate for the disturbance. Consider a pure-gain dynamic feedback-controlled process, as shown in Fig. 11.1. The process output can be expressed as

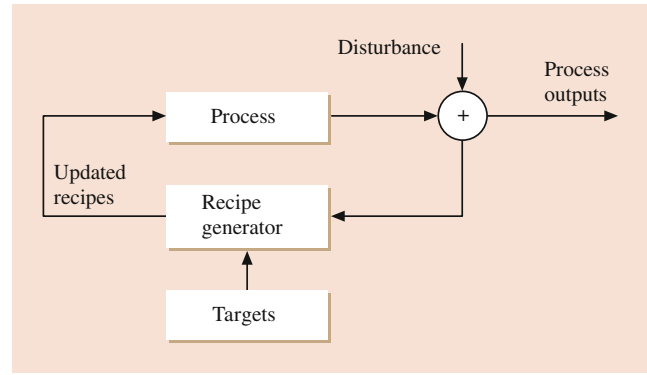


Fig. 11.1 Automatic process control

$$e_t = X_t - Z_{t-1}. \quad (11.7)$$

Suppose \hat{X}_t is an estimator (a predictor) of X_t that can be obtained at time $t - 1$. A realizable form of control can be obtained by setting

$$Z_{t-1} = -\hat{X}_t \quad (11.8)$$

so that the output error at time $t + 1$ becomes

$$e_t = X_t - \hat{X}_t, \quad (11.9)$$

which is equal to the "prediction error". Control and prediction can therefore have a one-to-one corresponding relationship via (11.8) and (11.9).

As shown in Box and Jenkins [45], when the process can be described by an ARIMA model, the MMSE control and the MMSE predictor have exactly the same form. Serving as an alternative to the MMSE predictor, the EWMA predictor corresponds to the integral (I) control [46] and is one of the most frequently used prediction methods due to its simplicity and efficiency. In general, the EWMA predictor is robust against nonstationarity due to the fact that the I control can continuously adjust the process whenever there is an offset.

An extension of the I control is the widely used PID control scheme,

$$Z_t = -k_p e_t - k_I \frac{1}{1-B} e_t - k_D (1-B) e_t, \quad (11.10)$$

where k_p, k_I , and k_D are constants that, respectively, determine the amount of proportional, integral, and derivative control action. The corresponding PID predictor (11.4) can be obtained from (11.8) and (11.10). When $\lambda_3 = 0$, in other words when $k_D = 0$ (and thus $\lambda_1 = k_p + k_I$ and $\lambda_2 = -k_p$), we have a PI predictor corresponding to the proportional-integral control scheme commonly used in industry.

Process Prediction Versus Forecast-Based Monitoring Methods

As discussed in Sect. 11.1.2, one class of SPC methods for autocorrelated processes starts from the idea of “whitening” the process and then monitoring the “whitened” process with time series prediction models. The SCC method monitors MMSE prediction errors and the M–M chart monitors the EWMA prediction error. Although the EWMA predictor is optimal for an IMA(0, 1, 1) process, the prediction error is no longer i.i.d. for predicting other processes. Most importantly, the EWMA prediction error that originated from the I control can compensate for mean shifts in steady state which makes the M–M chart very problematic for detecting small shifts in mean.

Since PID control is very efficient and robust, PID-based charts motivated by PID predictors outperform SCC and M–M charts. APC-based knowledge of the process can moreover clarify the performance of PID-based charts. In summary, the P term ensures that process output is close to the set point and thus sensitive in SPC monitoring, whereas the I term always yields control action regardless of error size which leads to a zero level of steady-state error. This implies that the I term is dominant in SPC monitoring. The purpose of derivative action in PID control is to improve closed-loop stability by making the D term in SPC monitoring less sensitive. Although there is no connection between the EWMA predictor and the EWMA chart, it is important to note that the I control leads to the EWMA predictor and the EWMA prediction-based chart is the M–M chart. As shown in Jiang et al. [37], the EWMA chart is the same as the P-based chart.

11.1.4 SPC for Automatically Controlled Processes

Although APC and SPC techniques share the objective of reducing process variation, their advocates have quarreled for decades. It has recently been recognized that the two techniques can be integrated to produce more efficient tools for process variation reduction [47–52]. This APC/SPC integration employs an APC rule to regulate the system and superimposes SPC charts on the APC-controlled system to detect process departures from the system model. Using Deming’s terminology, the APC scheme is responsible for reducing common cause variation, while the SPC charts are responsible for reducing assignable cause variation. From the statistical point of view, the former part resembles a parameter estimation problem for forecasting and adjusting the process and the latter part emulates a hypothesis test of process location. Figure 11.2 pictures a conceptual integration of SPC charts into the framework of a feedback control scheme. To avoid confusion, Box and Luceno [46] refer to

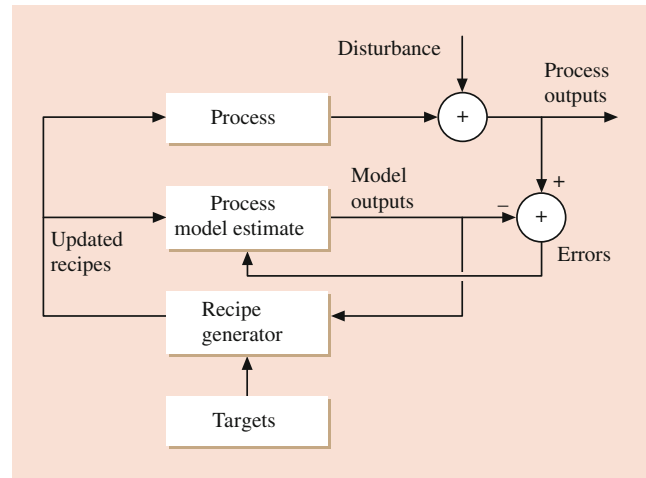


Fig. 11.2 APC/SPC integration

APC activities as process adjustment and to SPC activities as process monitoring. Since this chapter emphasizes SPC methods for quality improvement, we discuss only the monitoring component of APC/SPC integration.

As discussed in Sect. 11.1.3, control charts developed for monitoring autocorrelated observations shed light on the monitoring of integrated APC/SPC systems. Fundamentally, the output of an automatically controlled process is recommended for SPC monitoring. This is equivalent to forecast-based control charts of the corresponding predictor. For example, if the process is controlled by an MMSE controller, monitoring the output is exactly the same as the SCC method. Similar to forecast-based methods, assignable causes have an effect that is always contaminated by the APC control action which results in a limited “window of opportunity” for detection [35]. As an alternative, some authors suggest that monitoring the APC control action may improve the probability of detection [20]. Jiang and Tsui [53] compare the performance of monitoring the output vs. the control action of an APC process and show that for some autocorrelated processes monitoring the control action may be more efficient than monitoring the output of the APC system.

In general, the performance achieved by SPC monitoring an APC process depends on the data stream (the output or the control action) being measured, the APC control scheme employed, and the underlying autocorrelation of the process. If information from process output and control action can be combined, a universal monitor with higher SPC efficiency [51] can be developed. Kourti et al. [54] propose a method of monitoring process outputs conditional on the inputs or other changing process parameters. Li et al. [55] propose multivariate control charts such as Hotelling’s T^2 chart and the Bonferroni approach to monitor output and control action simultaneously. Defining the vector of outputs and control actions as $V_t = (e_t, \dots, e_{t-p+1}, X_t, \dots, X_{t-p+1})'$, a dynamic T^2

chart with window size p monitors statistic

$$W_t = V_t' \Sigma_V^{-1} V_t,$$

where Σ_V is the covariance matrix of V_t [56]. W_t follows a χ^2 distribution during each period given known process parameters. However, strong serial correlation exists so that the χ^2 quantiles cannot be used for control limits. By recognizing the mean shift patterns of V_t , Jiang [57] develops a GLRT procedure based on V_t . This GLRT procedure is basically univariate and more efficient than the T^2 chart.

11.1.5 Design of SPC Methods: Efficiency Versus Robustness

Among many others, the minimization of mean squared error/prediction error is one of the important criteria for prediction/control scheme design. Although the special cause chart is motivated by MMSE prediction/control, many previously mentioned SPC charts such as the PID chart have fundamentally different criteria from those of the corresponding APC controllers. When selecting SPC charts, the desired goal is maximization of the probability of shift detection.

For autocorrelated processes, Jiang [37] propose an ad hoc design procedure using PID charts. They demonstrate how two capability indices defined by signal-to-noise ratios (SNR) play a critical role in the evaluation of SPC charts. They denote σ_w as the standard deviation of charting statistic W_t and μ_T (μ_S) as the shift levels of W_t at the first step (l long enough) after the shift takes place. The transient state ratio is defined as $C_T = \mu_T/\sigma_w$, which measures the capability of the control chart to detect a shift in its first few steps. The steady state ratio is defined as $C_S = \mu_S/\sigma_w$, which measures the ability of the control chart to detect a shift in its steady state. These two signal-to-noise ratios determine the efficiency of the SPC chart and can be manipulated by selecting control chart parameters.

For a particular mean shift level, if the transient state ratio/capability can be tuned to a high value (say 4–5) by choosing appropriate chart parameters, the corresponding chart will detect the shift very quickly. Otherwise the shift will likely be missed during the transient state and will need to be detected in later runs. Since a high steady state ratio/capability heralds efficient shift detection at steady state, a high steady state ratio/capability is also desired. However, the steady state ratio/capability should not be tuned so high that it results in an extremely small transient ratio/capability, indicative of low probability of detection during the transient state. To endow the chart with efficient detection at both states, a tradeoff is needed when choosing the charting parameters. An approximate C_S value of 3 is generally appropriate for balancing the values of C_T and C_S .

One of the considerations when choosing an SPC method is its robustness to autocorrelated and automatically controlled processes. Robustness of a control chart refers to how insensitive its statistical properties are to model misspecification. Reliable estimates of process variation are of vital importance for the proper functioning of all SPC methods [58]. For process X_t with positive first-lag autocorrelation, the standard deviation derived from moving range is often underestimated because

$$E(\hat{\sigma}_{MR}) = E(\overline{MR}/d_2) = \sigma_X \sqrt{1 - \rho_1},$$

where ρ_1 is the first-lag correlation coefficient of X_t [59].

A more serious problem with higher sensitivity control charts such as the PID chart is that they may be less robust than lower sensitivity control charts such as the SCC. Berube et al. [60] and Luceno [61] conclude that PID controllers are generally more robust than MMSE controllers against model specification error. However Jiang [37] shows that PID charts tend to have a shorter “in-control” ARL when the process model is mis-specified since model errors can be viewed as a kind of “shift” from the “true” process model. This seems to be a discouraging result for higher sensitivity control charts. In practice, a trade-off is necessary between sensitivity and robustness when selecting control charts for autocorrelated processes. Apley and Lee [62] recommend using a conservative control limit for EWMA charts when monitoring MMSE residuals. By using the worst-case estimation of residual variance, the EWMA chart can be robustly designed for the in-control state with a slight efficiency loss in the out-of-control state. This design strategy can be easily generalized to other SPC methods for autocorrelated or automatically controlled processes.

11.1.6 SPC for Multivariate Characteristics

Through modern sensing technology that allows frequent measurement of key quality characteristics during manufacturing, many in-process measurements are strongly correlated to each other. This is especially true for measurements related to safety, fault detection and diagnosis, quality control, and process control. In an automatically controlled process, for example, process outputs are often strongly related to process control actions. Joint monitoring of these correlated characteristics ensures appropriate control of the overall process. Multivariate SPC techniques have recently been applied to novel fields such as environmental monitoring and detection of computer intrusion.

The purpose of multivariate on-line techniques is to investigate whether measured characteristics are simultaneously in statistical control. A specific multivariate quality control problem is to consider whether an observed vector of mea-

surements $\mathbf{x} = (x_1, \dots, x_k)$ exhibits a shift from a set of “standard” parameters $\boldsymbol{\mu}^0 = (\mu_1^0, \dots, \mu_k^0)'$. The individual measurements will frequently be correlated, meaning that their covariance matrix Σ will not be diagonal.

Versions of the univariate Shewhart, EWMA and CUSUM charts have been developed for the case of multivariate normality.

Multivariate T^2 Chart

To monitor a multivariate vector, *Hotelling* [63] suggested an aggregated statistic equivalent to the Shewhart control chart in the univariate case,

$$T^2 = (\mathbf{x} - \boldsymbol{\mu}^0)' \hat{\Sigma}_x^{-1} (\mathbf{x} - \boldsymbol{\mu}^0), \quad (11.11)$$

where $\hat{\Sigma}_x$ is an estimate of the population covariance matrix Σ . If the population covariance matrix is known, *Hotelling's* T^2 statistic follows a χ^2 distribution with k degrees of freedom when the process is in-control. A signal is triggered when $\chi^2 > \chi^2_{k,\alpha}$. One of the important features of the T^2 charts is that its out-of-control performance depends solely on the noncentrality parameter $\delta = \sqrt{(\boldsymbol{\mu} - \boldsymbol{\mu}^0)' \hat{\Sigma}_x^{-1} (\boldsymbol{\mu} - \boldsymbol{\mu}^0)}$, where $\boldsymbol{\mu}$ is the actual mean vector. This means that its detectional performance is invariant along the contours of the multivariate normal distribution.

Multivariate EWMA Chart

Hotelling's T^2 chart essentially utilizes only current process information. To incorporate recent historical information, *Lowry* [64] develop a similar multivariate EWMA chart

$$W_t^2 = \mathbf{w}_t' \Sigma_w^{-1} \mathbf{w}_t,$$

where $\mathbf{w}_t = \Lambda(\mathbf{x}_t - \boldsymbol{\mu}^0) + (\mathbf{I} - \Lambda)\mathbf{w}_{t-1}$ and $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_k)$. For simplicity, $\lambda_i = \lambda$ ($1 \leq i \leq k$) is generally adopted and $\Sigma_w = \lambda/(2 - \lambda)\Sigma_x$.

Multivariate CUSUM Chart

There are many CUSUM procedures for multivariate data. *Crosier* [65] proposes two multivariate CUSUM procedures, cumulative sum of T (COT) and MCUSUM. The MCUSUM chart is based on the statistics

$$s_t = \begin{cases} \mathbf{0} & \text{if } C_t \leq k_1 \\ (s_{t-1} + \mathbf{x}_t) (1 - k_1/C_t) & \text{if } C_t > k_1, \end{cases} \quad (11.12)$$

where $s_0 = \mathbf{0}$, $C_t = \sqrt{(s_{t-1} + \mathbf{x}_t)' \Sigma_x^{-1} (s_{t-1} + \mathbf{x}_t)}$, and $k_1 > 0$. The MCUSUM chart signals when $W_t = s_t' \Sigma_x^{-1} s_t > h_1$. *Pignatiello* and *Runger* [66] propose another multivariate CUSUM chart (MC1) based on the vector of cumulative sums,

$$W_t = \max\left(0, \sqrt{D_t' \Sigma_x^{-1} D_t} - k_2 l_t\right) \quad (11.13)$$

where $k_2 > 0$, $D_t = \sum_{i=t-l_t+1}^t \mathbf{x}_i$, and

$$l_t = \begin{cases} l_{t-1} + 1 & \text{if } W_{t-1} > 0 \\ 1 & \text{otherwise.} \end{cases}$$

Once an out-of-control signal is triggered from a multivariate control chart, it is important to track the cause of the signal so that the process can be improved. Fault diagnosis can be implemented by T^2 decompositions following the signal and large components are suspected to be faulty. Orthogonal decompositions such as principal component analysis are popular tools. *Hayter* and *Tsui* [67] propose other alternatives which integrate process monitoring and fault diagnosis. *Jiang* and *Tsui* [68] provide a thorough review of these methods.

Variable-Selection-Based Multivariate Chart

In high dimensional applications, it is very rare to see all interested variables or quality characteristics change or shift at the same time. Rather, a typical yet common phenomenon observed in practice is that a subset of variables, which is dominated by a common latent physical mechanism or component, deviate from their normal condition due to abnormal changes of the common mechanism or component [69–71]. By penalizing likelihood functions to locate potential out-of-control variables, *Wang* and *Jiang* [72] and *Zou* and *Qiu* [73] independently propose to monitor a variable-dimension T^2 statistic, which has better efficiency than traditional full-size T^2 statistic. *Zou* et al. [74] and *Jiang* et al. [75] further utilized the LASSO algorithm for fault diagnosis.

Multivariate Chart Using Real-Time Contrast

Instead of monitoring departures from a nominal mean vector in Phase II, multivariate RTC control charts monitor distances between real time data and Phase I reference data using classification methods. Mis-classification probabilities serve as a reasonable candidate for monitoring differences between the two populations [76]. Classification methods such as linear discrimination analysis (LDA), support vector machines (SVM), etc. can be deployed and kernel-based methods can also be adapted to account for nonlinear boundary between Phases I and II data [77, 78]. Since these classification methods look for a projection direction such that certain “distance” metric are optimized, projection pursuit can be generalized by measuring empirical divergence between the two probability distributions for real-time monitoring [79].

11.1.7 SPC for Profile Monitoring

In many applications, the quality of a process or product is best characterized and summarized by a functional relationship between a response variable and one or more explanatory variables. Profile monitoring is used to understand and to check the stability of this relationship over time. At each sampling stage one observes a collection of data points that can be represented by a curve (or profile). In some calibration applications, the profile can be represented adequately by a simple linear regression model, while in other applications more complicated models are needed.

Profile monitoring is very useful in an increasing number of practical applications. Much of the work in the past few years has focused on the use of more effective charting methods, the study of more general shapes of profiles, and the study of the effects of violations of assumptions. There are many promising research topics yet to be pursued given the broad range of profile shapes and possible models. Woodall et al. (2004) [80] highlighted the following important issues when monitoring profiles:

1. The usefulness of carefully distinguishing between Phase I and Phase II applications.
2. The decision regarding whether or not to include some between profile variation in common cause variation.
3. The use of methods capable of detecting any type of shift in the shape of the profile.
4. The use of the simplest adequate profile model.

Paynabar et al. [81] developed a new modeling, monitoring, and diagnosis framework for phase-I analysis of multichannel profiles. Woodall et al. [82] conducted a comprehensive review on the use of control charts to monitor process and product quality profiles.

11.2 Design of Experiment and Robust Parameter Design

11.2.1 Robust Design for Single Responses

Taguchi [4] introduced parameter design, a method for designing processes that are robust (insensitive) to uncontrollable variation, to a number of American corporations. The objective of this methodology is to find the settings of design variables that minimize the expected value of squared-error loss defined as

$$L(Y, t) = (Y - t)^2, \quad (11.14)$$

where Y represents the actual process response and t the targeted value. A loss occurs if the response Y deviates from

its target t . This loss function originally became popular in estimation problems considering unbiased estimators of unknown parameters. The expected value of $(Y - t)^2$ can be easily expressed as

$$\begin{aligned} E(L) &= A_0 E(Y - t)^2 \\ &= A_0 [\text{Var}(Y) + (E(Y) - t)^2], \end{aligned} \quad (11.15)$$

where $\text{Var}(Y)$ and $E(Y)$ are the mean and variance of the process response and A_0 is a proportional constant representing the economic costs of the squared error loss. If $E(Y)$ is on target then the squared-error loss function reduces to the process variance. Its similarity to the criterion of least squares in estimation problems makes the squared-error loss function easy for statisticians and engineers to grasp. Furthermore the calculations for most decision analyses based on squared-error loss are straightforward and easily seen as a trade-off between variance and the square of the off-target factor.

Robust design (RD) assumes that the appropriate performance measure can be modeled as a transfer function of the fixed control variables and the random noise variables of the process as follows:

$$Y = f(\mathbf{x}, \mathbf{N}, \boldsymbol{\theta}) + \epsilon, \quad (11.16)$$

where $\mathbf{x} = (x_1, \dots, x_p)^T$ is the vector of control factors, $\mathbf{N} = (N_1, \dots, N_q)^T$ is the vector of noise factors, $\boldsymbol{\theta}$ is the vector of unknown response model parameters, and f is the transfer function for Y . The control factors are assumed to be fixed and represent the fixed design variables. The noise factors \mathbf{N} are assumed to be random and represent the uncontrolled sources of variability in production. The pure error ϵ represents the remaining variability that is not captured by the noise factors and is assumed to be normally distributed with zero mean and finite variance.

Taguchi divides the design variables into two subsets, $\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_d)$, where \mathbf{x}_a and \mathbf{x}_d are called respectively the adjustment and nonadjustment design factors. An adjustment factor influences process location while remaining effectively independent of process variation. A nonadjustment factor influences process variation.

Experimental Designs for Parameter Design

Taguchi's Product Arrays and Combined Arrays

Taguchi's experimental design takes an orthogonal array for the controllable design parameters (an inner array of control factors) and crosses it with another orthogonal array for the factors beyond reasonable control (an outer array of noise factors). At each test combination of control factor levels, the entire noise array is run and a performance measure is calculated. Hereafter we refer to this design as the product

array. These designs have been criticized by *Box* [12] and others for being unnecessarily large.

Welch [83] combined columns representing the control and noise variables within the same orthogonal array. These combined arrays typically have a shorter number of test runs and do not replicate the design. The lack of replication prevents unbiased estimation of random error but we will later discuss research addressing this limitation.

Which to Use: Product Array or Combined Array

There is a wide variety of expert opinion regarding choice of experimental design in *Nair* [13]. The following references complement *Nair*'s comprehensive discussion. *Ghosh and Derderian* [84] derive robustness measures for both product and combined arrays, allowing the experimenter to objectively decide which array provides a more robust option. *Miller et al.* [85] consider the use of a product array on gear pinion data. *Lucas* [86] concludes that the use of classical, statistically designed experiments can achieve the same or better results than Taguchi's product arrays. *Rosenbaum* [87] reinforces the efficiency claims of the combined array by giving a number of combined array designs which are smaller for a given orthogonal array strength or stronger for a given size. Finally, *Wu and Hamada* [88] provide an intuitive approach to choosing between product and combined array based on an effect-ordering principle.

They list the most important class of effects as those containing control–noise interactions, control main effects and noise main effects. The second highest class contains the control–control interactions and the control–control–noise interactions while the third and least important class contains the noise–noise interactions. That array producing the highest number of clear effect estimates in the most important class is considered the best design.

Noting that the combined array is often touted as being more cost-effective due to an implied smaller number of runs, *Wu and Hamada* place the cost comparison on a more objective basis by factoring in both cost per control setting and cost per noise replicate. They conclude that the experimenter must prioritize the effects to be estimated and the realistic costs involved before deciding which type of array is optimal.

Choosing the Right Orthogonal Array for RD

Whether the experimenter chooses a combined or product array, selecting the best orthogonal array is an important consideration. The traditional practice in classical design of experiments is to pick a Resolution IV or higher design so that individual factors are aliased with three factor interactions, of which there are relatively few known physical examples.

However, the estimation of main effects is not necessarily the best way to judge the value of a test design for RD. The control–noise interactions are generally regarded as having equal importance as the control effects for fine tuning the

final control factor settings for minimal product variation. Hence evaluation of an experimental design for RD purposes must take into account the design's ability to estimate the control–noise interactions deemed most likely to affect product performance.

Kackar and Tsui [89] feature a graphical technique for showing the confounding pattern of effects within a two-level fractional factorial. *Kackar et al.* [90] define orthogonal arrays and describe how Taguchi's fixed element arrays are related to well known fractional factorial designs. Other pieces related to this decision are *Hou and Wu* [91], *Berube and Nair* [60], and *Bingham and Sitter* [92].

D-Optimal Designs

In this section several authors show how D-optimal designs can be exploited in RD experiments. A D-optimal design minimizes the area of the confidence ellipsoids for parameters being estimated from an assumed model. Their key strength is their invariance to linear transformation of model terms and their characteristic weakness is a dependence on the accuracy of the assumed model. By using a proper prior distribution to attack the singular design problem and make the design less model-dependent, *Dumouchel and Jones* [93] provide a Bayesian D-optimal design needing little modification of existing D-optimal search algorithms.

Atkinson and Cook [94] extend the existing theory of D-optimal design to linear models with nonconstant variance. With a Bayesian approach they create a compromise design that approximates preposterior loss. *Vining and Schaub* [95] use D-optimality to evaluate separate linear models for process mean and variance. Their comparison of the designs indicates that replicated fractional factorials of assumed constant variance best estimate variance while semi-Bayesian designs better estimate process response.

Chang [96] proposes an algorithm for generating near D-optimal designs for multiple response surface models. This algorithm differs from existing approaches in that it does not require prior knowledge or data-based estimates of the covariance matrix to generate its designs. *Mays* [97] extends the quadratic model methodology of RSM to the case of heterogeneous variance by using the optimality criteria D (maximal determinant) and I (minimal integrated prediction variance) to allocate test runs to locations within a central composite design.

Other Designs

The remaining references discuss types of designs used in RD which are not easily classified under the more common categories previously discussed.

Pledger [98] divides noise variables into observable and unobservable and argues that one's ability to observe selected noise variables in production should translate into better choices of optimal control settings. *Rosenbaum* [99] uses blocking to separate the control and noise variables in

combined arrays, which were shown in *Rosenbaum* [87] to be stronger for a given size than the corresponding product array designs. *Li* and *Nachtsheim* [100] present experimental designs which don't depend on the experimenter's prior determination of which interactions are most likely significant.

Performance Measures in RD

In Sect. 11.2.1, we compared some of the experimental designs used in parameter design. Of equal importance is choosing which performance measure will best achieve the desired optimization goal.

Taguchi's Signal-to-Noise Ratios

Taguchi introduced a family of performance measures called signal-to-noise ratios whose specific form depends on the desired response outcome. The case where the response has a fixed nonzero target is called the nominal-the-best case (NTB). Likewise, the cases where the response has a smaller-the-better target or a larger-the-better target are, respectively, called the STB and LTB cases.

To accomplish the objective of minimal expected squared-error loss for the NTB case, Taguchi proposed the following two-step optimization procedure: (i) Calculate and model the SNRs and find the nonadjustment factor settings which maximize the SNR. (ii) Shift mean response to the target by changing the adjustment factor(s).

For the STB and LTB cases, Taguchi recommends directly searching for the values of the design vector \mathbf{x} which maximize the respective SNR. Alternatives for these cases are provided by *Tsui* and *Li* [101] and *Berube* and *Wu* [102].

Performance Measure Independent of Adjustment (PerMIAs)

Taguchi did not demonstrate how minimizing the SNR would achieve the stated goal of minimal average squared-error loss. *Leon* et al. [11] defined a function called the performance measure independent of adjustment (PerMIA) which justified the use of a two-step optimization procedure. They also showed that Taguchi's SNR for the NTB case is a PerMIA when both an adjustment factor exists and the process response transfer function is of a specific multiplicative form. When Taguchi's SNR complies with the properties of a PerMIA, his two-step procedure minimizes the squared-error loss.

Leon et al. [11] also emphasized two major advantages of the two-step procedure:

- It reduces the dimension of the original optimization problem.
- It does not require reoptimization for future changes of the target value.

Box [12] agrees with *Leon* et al. [11] that the SNR is only appropriately used in concert with models where process sigma is proportional to process mean. *Maghsoodloo* [103] derives and tabulates exact mathematical relationships between Taguchi's STB and LTB measures and his quality loss function.

Leon and *Wu* [104] extend the PerMIA of *Leon* et al. [11] to a maximal PerMIA which can solve constrained minimization problems in a two-step procedure similar to that of Taguchi. For nonquadratic loss functions, they introduce general dispersion, location, and off-target measures, while developing a two-step process. They apply these new techniques in a number of examples featuring additive and multiplicative models with nonquadratic loss functions. *Tsui* and *Li* [101] establish a multistep procedure for the STB and LTB problem based on the response model approach under certain conditions.

Process Response and Variance as Performance Measures

The dual response approach is a way of finding the optimal design settings for a univariate response without the need to use a loss function. Its name comes from its treatment of mean and variance as responses of interest which are individually modeled. It optimizes a primary response while holding the secondary response at some acceptable value.

Nair and *Pregibon* [105] suggest using outlier-robust measures of location and dispersion such as median (location) and interquartile range (dispersion). *Vining* and *Myers* [106] applied the dual response approach to Taguchi's three SNRs while restricting the search area to a spherical region of limited radius. *Copeland* and *Nelson* [107] solve the dual response optimization problem with the technique of direct function minimization. They use the Nelder-Mead simplex procedure and apply it to the LTB, STB, and NTB cases. Other noteworthy papers on the dual response method include *Del Castillo* and *Montgomery* [108] and *Lin* and *Tu* [109].

Desirability as a Performance Measure

The direct conceptual opposite of a loss function, a utility function maps a specific set of design variable settings to an expected utility value (value or worth of a process response). Once the utility function is established, nonlinear direct search methods are used to find the vector of design variable settings that maximizes utility.

Harrington [110] introduced a univariate utility function called the desirability function, which gives a quality value between zero (unacceptable quality) and one (further improvement would be of no value) of a quality characteristic of a product or process. He defined the two-sided desirability function as follows:

$$d_i = e^{-|Y_i|^c}, \quad (11.17)$$

where e is the natural logarithm constant, c is a positive number subjectively chosen for curve scaling, and Y_i' is a linear transformation of the univariate response Y_i whose properties link the desirability values to product specifications. It is of special interest to note that for $c = 2$, a mid-specification target and response values within the specification limits, this desirability function is simply the natural logarithm constant raised to the squared-error loss function.

Other Performance Measures

Ng and *Tsui* [111] derive a measure called q-yield which accounts for variation from target among passed units as well as nonconforming units. It does this by penalizing yield commensurate with the amount of variation measured within the passed units. *Joseph* and *Wu* [102] develop modeling and analysis strategies for a general loss function where the quality characteristic follows a location-scale model. Their three-step procedure includes an adjustment step which moves the mean to the side of the target with lower cost. Additional performance measures are introduced in *Joseph* and *Wu* [112] and *Joseph* and *Wu* [113].

Modeling the Performance Measure

The third important decision the experimenter must grapple with is how to model the chosen performance measure. Linear models are by far the most common way to approximate loss functions, SNR's, and product responses. This section covers response surface models, the generalized linear model, and Bayesian modeling.

Response Surface Models

Response surface models (RSM) are typically second-order linear models with interactions between the first-order model terms. While many phenomena cannot be accurately represented by a quadratic model, the second-order approximation of the response in specific regions of optimal performance may be very insightful to the product designer.

Myers et al. [114] make the case for implementing Taguchi's philosophy within a well established, sequential body of empirical experimentation, RSM. The combined array is compared to the product array and the modeling of SNR compared to separate models for mean and variance. In addition, RSM lends itself to the use of mixed models for random noise variables and fixed control variables. *Myers* et al. [115] incorporate noise variables and show how mean and variance response surfaces can be combined to create prediction limits on future response.

Analysis of Unreplicated Experiments

The most commonly cited advantage of modeling process responses rather than SNR is the use of more efficient

combined arrays. However, the gain in efficiency usually assumes there is no replication for estimating random error. Here we review references for analyzing the data from unreplicated fractional factorial designs.

Box and *Meyer* [116] present an analysis technique which complements normal probability plots for identifying significant effects from an unreplicated design. Their Bayesian approach assesses the size of contrasts by computing a posterior probability that each contrast is active. They start with a prior probability of activity and assume normality of the significant effects and deliver a nonzero posterior probability for each effect.

Lenth [117] introduces a computationally simple and intuitively pleasing technique for measuring the size of contrasts in unreplicated fractional factorials. The Lenth method uses standard T statistics and contrast plots to indicate the size and significance of the contrast. Because of its elegant simplicity, the method of Lenth is commonly cited in RD case studies.

Pan [118] shows how failure to identify even small and moderate location effects can subsequently impair the correct identification of dispersion effects when analyzing data from unreplicated fractional factorials. *Wu* and *Hamada* [88] propose a simple simulation method for estimating the critical values employed by Lenth in his method for testing significance of effects in unreplicated fractional factorial designs.

McGrath and *Lin* [119] show that a model that does not include all active location effects raises the probability of falsely identifying significant dispersion factors. They show analytically that without replication it is impossible to deconfound a dispersion effect from two location effects.

Generalized Linear Model

The linear modeling discussed in this chapter assumes normality and constant variance. When the data does not demonstrate these properties, the most common approach is to model a compliant, transformed response. In many cases, this is hard or impossible. The general linear model (GLM) was developed by *Nelder* and *Wedderburn* [120] as a way of modeling data whose probability distribution is any member of the single parameter exponential family.

The GLM is fitted by obtaining the maximum likelihood estimates for the coefficients to the terms in the linear predictor, which may contain continuous, categorical, interaction, and polynomial terms. *Nelder* and *Lee* [121] argue that the GLM can extend the class of useful models for RD experiments to data-sets, wherein a simple transformation cannot necessarily satisfy the important criteria of normality, separation, and parsimony. Several examples illustrate how the link functions are chosen.

Engel and *Huele* [122] integrate the GLM within the RSM approach to RD. Nonconstant variance is assumed and models for process mean and variance are obtained from a heteroscedastic linear model of the conditional process

response. The authors claim that nonlinear models and tolerances can also be studied with this approach. *Hamada and Nelder* [123] apply the techniques described in *Nelder and Lee* [121] to three quality improvement examples to emphasize the utility of the GLM in RD problems over its wider class of distributions.

Bayesian Modeling

Bayesian methods of analysis are steadily finding wider employment in the statistical world as useful alternatives to frequentist methods. In this section we mention several references on Bayesian modeling of the data.

Using a Bayesian GLM, *Chipman and Hamada* [124] overcome the GLM's potentially infinite likelihood estimates from categorical data taken from fractional factorial designs. *Chipman* [125] uses the model selection methodology of *Box and Meyer* [126] in conjunction with priors for variable selection with related predictors. For optimal choice of control factor settings, he finds posterior distributions to assess the effect of model and parameter uncertainty.

11.2.2 Robust Design for Multiple Responses

Earlier we discussed loss and utility functions and showed how the relation between off-target and variance components underlies the loss function optimization strategies for single responses. Multi-response optimization typically combines the loss or utility functions of individual responses into a multivariate function to evaluate the sets of responses created by a particular set of design variable settings. This section is divided into two subsections which, respectively, deal with the additive and multiplicative combination of loss and utility functions, respectively.

Additive Combination of Univariate Loss, Utility and SNR

The majority of multiple response approaches additively combine the univariate loss or SNR performance measures discussed. In this section, we review how these performance measures are additively combined and their relative advantages and disadvantages as multivariate objective functions.

Multivariate Quadratic Loss

For univariate responses, expected squared-error loss is a convenient way to evaluate the loss caused by deviation from target because of its decomposition into squared off-target and variance terms. A natural extension of this loss function to multiple correlated responses is the multivariate quadratic loss (MQL) function of the deviation vector $(Y - \tau)$ where $Y = (Y_1, \dots, Y_r)^T$ and $\tau = (t_1, \dots, t_r)^T$, i.e.,

$$\text{MQL}(Y, \tau) = (Y - \tau)^T A (Y - \tau), \quad (11.18)$$

where A is a positive definite constant matrix. The values of the constants in A are related to the costs of nonoptimal design, such as the costs related to repairing and/or scrapping noncompliant product. In general, the diagonal elements of A represent the weights of the r characteristics and the off-diagonal elements represent the costs related to pairs of responses being simultaneously off-target.

It can be shown that, if Y follows a multivariate normal distribution with mean vector $E(Y)$ and covariance matrix Σ_Y , the average (expected) loss can be written as:

$$\begin{aligned} E(\text{MQL}) &= E(Y - \tau)^T A (Y - \tau) \\ &= \text{Tr}(A \Sigma_Y) \\ &\quad + [E(Y) - \tau]^T A [E(Y) - \tau]. \end{aligned} \quad (11.19)$$

The simplest approach to solving the RD problem is to apply algorithms to directly minimize the average loss function in (11.19). Since the mean vector and covariance matrix are usually unknown, they can be estimated by the sample mean vector and sample covariance matrix or a fitted model based on a sample of observations of the multivariate responses. The off-target vector product $[E(Y) - \tau]^T A [E(Y) - \tau]$ and $\text{Tr}(A \Sigma_Y)$ are multivariate analogs to the squared off-target component and variance of the univariate squared-error loss function. This decomposition shows how moving all response means to target simplifies the expected multivariate loss to the $\text{Tr}(A \Sigma_Y)$ term. The trace-covariance term shows how the values of A and the covariance matrix Σ_Y directly affect the expected multivariate loss.

Optimization of Multivariate Loss Functions

For the expected multivariate quadratic loss of (11.19), *Pignatiello* [16] introduced a two-step procedure for finding the design variable settings that minimize this composite cost of poor quality. *Tsui* [18] extended *Pignatiello's* two-step procedure to situations where responses may be NTB, STB or LTB.

To this point we have examined squared-error loss functions whose expected value is decomposed into off-target and variance components. *Ribeiro and Elsayed* [127] introduced a multivariate loss function which additionally considers fluctuation in the supposedly fixed design variable settings. *Ribeiro et al.* [128] add a term for manufacturing cost to the gradient loss function of *Ribeiro and Elsayed*.

Additive Formation of Multivariate Utility Functions

Kumar et al. [129] suggest creating a multiresponse utility function as the additive combination of utility functions from the individual responses where the goal is to find the set of design variable settings that maximizes overall utility. Additional papers related to this technique include *Artiles-Leon* [130] and *Ames et al.* [131].

Quality Loss Functions for Nonnegative Variables

Joseph [132] argues that, in general, processes should not be optimized with respect to a single STB or LTB characteristic, rather than a combination of them. He introduces a new class of loss functions for nonnegative variables which accommodates the cases of unknown target and asymmetric loss and which can be additively combined for the multiresponse case.

Multivariate Utility Functions from Multiplicative Combination

In this section, a multivariate desirability function is constructed from the geometric average of the individual desirability functions of each response.

The geometric average (GA) of r components (d_1, \dots, d_r) is the r th root of their products:

$$\text{GA}(d_1, \dots, d_r) = \left(\prod_{i=1}^r d_i \right)^{\frac{1}{r}}. \quad (11.20)$$

The GA is then a multiplicative combination of the individuals. When combining individual utility functions whose values are scaled between zero and one, the GA yields a value less than or equal to the lowest individual utility value. When rating the composite quality of a product, this prevents any single response from reaching an unacceptable value, since a very low value on any crucial characteristic (such as safety features or cost) will render the entire product worthless to the end user.

Modifications of the Desirability Function

In order to allow placement of the ideal target value anywhere within the specifications, *Derringer* and *Suich* [133] introduced a modified version of the desirability functions of *Harrington* [110] which encompassed both one-sided and two-sided response specifications. Additional extensions of the multivariate desirability function were made by *Kim* and *Lin* [134].

Alternative Performance Measures for Multiple Responses

Duffy et al. [135] propose using a reasonably precise estimate of multivariate yield, obtained via Beta distribution discrete point estimation, as an efficient alternative to Monte Carlo simulation. This approach is limited to independently distributed design variables. *Fogliatto* and *Albin* [136] propose using predictor variance as a multiresponse optimization criterion. They measure predictive variance as the coefficient of variance (CV) of prediction since it represents a normalized measure of prediction variance. *Plante* [137] considers the use of maximal process capability as the criterion for choosing control variable settings in multiple response RD situations. He uses the concepts of process capability and

desirability to develop process capability measures for multiple response systems.

11.2.3 Dynamic Robust Design

Taguchi's Dynamic Robust Design

Up to this point, we've discussed only static RD, where the targeted response is a given, fixed level and is only affected by control and noise variables. In dynamic robust design (DRD) a third type of variable exists, the signal variable M whose magnitude directly affects the mean value of the response. The experimental design recommended by Taguchi for DRD is the product array consisting of an inner control array crossed with an outer array consisting of the sensitivity factors and a compound noise factor.

A common choice of dynamic loss function is the quadratic loss function popularized by Taguchi,

$$L[Y, t(M)] = A_0[Y - t(M)]^2, \quad (11.21)$$

where A_0 is a constant. This loss function provides a good approximation to many realistic loss functions. It follows that the average loss becomes

$$\begin{aligned} R(\mathbf{x}) &= A_0 E_M E_{N,\epsilon} [Y - t(M)]^2 \\ &= A_0 E_M \left\{ \text{Var}_{N,\epsilon}(Y) + [E_{N,\epsilon}(Y) - t(M)]^2 \right\}. \end{aligned} \quad (11.22)$$

Taguchi identifies dispersion and sensitivity effects by modeling SNR respectively as a function of control factors and sensitivity factors. His two-step procedure for DRD finds control factor settings to minimize SNR and sets other, non-SNR related control variables to adjust the process to the targeted sensitivity level.

References on Dynamic Robust Design

Ghosh and *Derderian* [138] introduce the concept of robustness of the experimental plan itself to the noise factors present when conducting DRD. For combined arrays they consider blocked and split-plot designs and for product arrays they consider univariate and multivariate models. In product arrays they do this by choosing settings which minimize the noise factor effects on process variability and for the combined array they attempt to minimize the interaction effects between control and noise factors.

Wasserman [139] clarifies the use of the SNR for the dynamic case by explaining it in terms of linear modeling of process response. He expresses the dynamic response as a linear model consisting of a signal factor, the true sensitivity (β) at specific control variable settings, and an error term. *Miller* and *Wu* [140] prefer the term signal-response system to dynamic robust design for its intuitive appeal and its iden-

tification of two distinct types of signal-response systems. They call them measurement systems and multiple target systems, where this distinction determines the performance measure used to find the optimal control variable settings.

Lunani et al. [141] present two new graphical procedures for identifying suitable measures of location and dispersion in RD situations with dynamic experimental designs. *McCaskey and Tsui* [142] show that Taguchi's two-step procedure for dynamic systems is only appropriate for multiplicative models and develop a procedure for dynamic systems under an additive model. For a dynamic system this equates to minimizing the sum of process variance and bias squared over the range of signal values.

Tsui [143] compares the effect estimates obtained using the response model approach and Taguchi's approach for dynamic robust design problems. Recent publications on DRD include *Joseph and Wu* [144], *Joseph and Wu* [145], and *Joseph* [146].

11.2.4 Applications of Robust Design

Manufacturing Case Studies

Mesenbrink [147] applied the techniques of RD to optimize three performance measurements of a high volume wave soldering process. They achieved significant quality improvement using a mixed-level fractional factorial design to collect ordered categorical data regarding the soldering quality of component leads in printed circuit boards. *Lin and Wen* [148] apply RD to improve the uniformity of a zinc coating process.

Chhajer and Lowe [149] apply the techniques of RD to the problem of structured tool management. For the cases of tool selection and tool design, they use Taguchi's quadratic loss function to find the most cost effective way to accomplish the processing of a fixed number of punched holes in sheet metal products.

Reliability Applications

Reliability is the study of how to make products and processes function for longer periods of time with minimal interruption. It is a natural area for RD application and the Japanese auto industry has made huge strides in this area compared to its American counterpart. In this section several authors comment on the application of RD to reliability.

Hamada [150] demonstrates the relevance of RD to reliability improvement. He recommends the response model approach for the additional information it provides on control-noise interactions and suggests alternative performance criteria for maximizing reliability. *Kuhn et al.* [151] extend the methods of *Myers et al.* [114] for linear models and normally distributed data to achieve a robust process when time to an event is the response.

Tolerance Design

This chapter has focused on RD, which is synonymous with Taguchi's methods of parameter design. Taguchi has also made significant contributions in the area of tolerance design. This section reviews articles which examine developments in the techniques of tolerance design.

D'errico and Zaino [152] propose a modification of Taguchi's approach to tolerance design based on a product Gaussian quadrature which provides better estimates of high-order moments and outperforms the basic Taguchi method in most cases. *Bisgaard* [153] proposes using factorial experimentation as a more scientific alternative to trial and error to design tolerance limits when mating components of assembled products.

Zhang and Wang [154] formulate the robust tolerance problem as a mixed nonlinear optimization model and solve it using a simulated annealing algorithm. The optimal solution allocates assembly and machining tolerances so as to maximize the product's insensitivity to environmental factors. *Li and Wu* [55] combined parameter design with tolerance design.

Maghsoodloo and Li [155] consider linear and quadratic loss functions for determining an optimal process mean which minimizes the expected value of the quality loss function for asymmetric tolerances of quality characteristics. *Moskowitz et al.* [156] develop parametric and nonparametric methods for finding economically optimal tolerance allocations for a multivariable set of performance measures based on a common set of design parameters.

11.3 Reliability and Prognostics and Health Management

11.3.1 Prognostics and Health Management

Prognostics and health management (PHM) is a framework that offers comprehensive solutions for monitoring and managing health status of individual machine and engineering systems. In recent years, PHM has emerged to be a popular approach for improving reliability, maintainability, safety, and affordability. Concepts and components in PHM have been applied in many domain areas such as mechanical engineering, electrical engineering, statistical science, etc.

Due to high impact and extreme costs associated with system failures, it is important to develop methods that can predict and prevent such catastrophes before they occur. Many application methods have been developed in domains such as electronics-rich systems, aerospace industries, or even the public health environment [157, 158], which can be grouped under the framework of prognostics and health management (PHM). Prognostics is the process of predicting the future reliability of a product by assessing its degradation

from its expected normal operating conditions; health management is the process of real time monitoring the extent of deviation or degradation from normal operating condition [159, 160]. Traditional reliability prediction methods (e.g., US Department of Defense Mil-Hdbk-217 and Telcordia SR-332 (formerly [161])), make strong assumptions that constant hazard rate of each component can be modified to account for various operating and environmental conditions. In PHM approach, we monitor the system's health status in real time and dynamically update the reliability and hazard function based on in situ measurements and update the current models based on historical data. Due to the success of the PHM approach, new PHM techniques and methods are needed and to apply and implement PHM to other and underdeveloped domains.

Due to the increasing complexity of modern systems, one most prominent problem is called the No Fault Found (NFF) problem (similarly, "trouble not identified," "intermittent malfunctions," etc.) [162–164], particularly in electronics-rich systems. It refers to the situation that no failure or fault can be detected or duplicated during laboratory tests even when the failure has been reported in the field [165]. NFF issues not only make diagnosis extremely difficult but also result in skyrocketing maintenance costs. As reported by Williams et al. [165], NFF failures account for more than 85% or 90% of all filed failures and overall maintenance costs in avionics respectively, which cost the US Department of Defense roughly 2 ~ 10 billion US dollars annually [166]. Similarly, NFF contributes to significant operational costs in many other domain areas. In addition, NFF contributes to potential safety hazards in other industries. For example, both Toyota and National Highway Traffic Safety Administration (NHTSA) spent quite a time and efforts to investigate the root causes of sudden acceleration failures in certain car models, which is linked to 89 deaths in 71 crashes since 2000 [167]. Unfortunately, no conclusive finding has been reached despite efforts to replicate the failures in a variety of laboratory conditions. From these examples, it was found that intermittent faults are often related to environmental conditions and operation histories of the particular individual system, and thus difficult to duplicated repeated under newly unknown random disturbances. Traditional laboratory testing and assessment data only provide information on the characteristics of the population and are insufficient to lead accurate prediction for each individual performance. To reduce maintenance cost and avoid safety hazards caused by NFF, the approach of PHM shifts from traditional population data modeling from individual data modeling.

In response to these challenges, the fast development of information and sensing technology has enabled the collection of many in situ measurements during operations that provides the capability of real time data management and processing for each individual. These advancements provide

a great opportunity to develop sophisticated models with increasing accuracy of prognostics for individual items. For instance, many different types of data during the whole life cycle of the products can be easily retrieved, especially in critical applications. These data could include production process information, quality records, operation logs, and sensor measurements. Moreover, unlike manually entered data, which are slow, costly, and error-prone, many current records are accurate and timely due to advancements in automated technology. The use of Radio Frequency Identification (RFID) technology, for example, is commonly used in supply chain distribution networks, healthcare, and even military applications, because it provides reliable and timely tracking of products/components. Advanced sensor technologies also enable abundant measurements at both macro and micro scales, such as those used to measure vibration, frequency response, magnetic fields, and the current/voltage, to name a few.

In general, typical workflows in a PHM system can be conceptually divided into three major tasks: fault diagnostics, prognostics, and condition-based maintenance. The first task is on diagnose and root causes identification for system failures. The root causes provide useful information for prognostic and feedback for system design improvement. The prognostic task takes the processed data, system models, or failure mode analysis as inputs and employs the developed prognosis algorithms to update the degradation models for failure time prediction. The last task makes use of the prognosis results with consideration of the cost and benefits to determine the optimal maintenance actions to achieve minimal operating costs and risks. All of these three tasks are necessarily executed dynamically and in real time.

11.3.2 Systems Health Monitoring and Management

Systems health monitoring and management (SHMM) refers to the framework of continuous surveillance, analysis, and interpretation of relevant data for system maintenance, management, and strategic planning, where "system" is generally defined as "an organized set of detailed methods, procedures, and routines created to carry out a specific activity or solve a specific problem," ranging from mechanical systems to public health [168–170]. SHMM differs from PHM by its distinct emphasis and its definitions of monitoring, prognostics, and management, and can be considered an extended version of PHM. More specifically, system health monitoring includes detection, forecasting, diagnostics, and prognostics, while system health management includes decision, financial, and risk management. A fundamental problem in SHMM is on how to make use of correlated active and passive data in various tasks of prediction and forecasting,

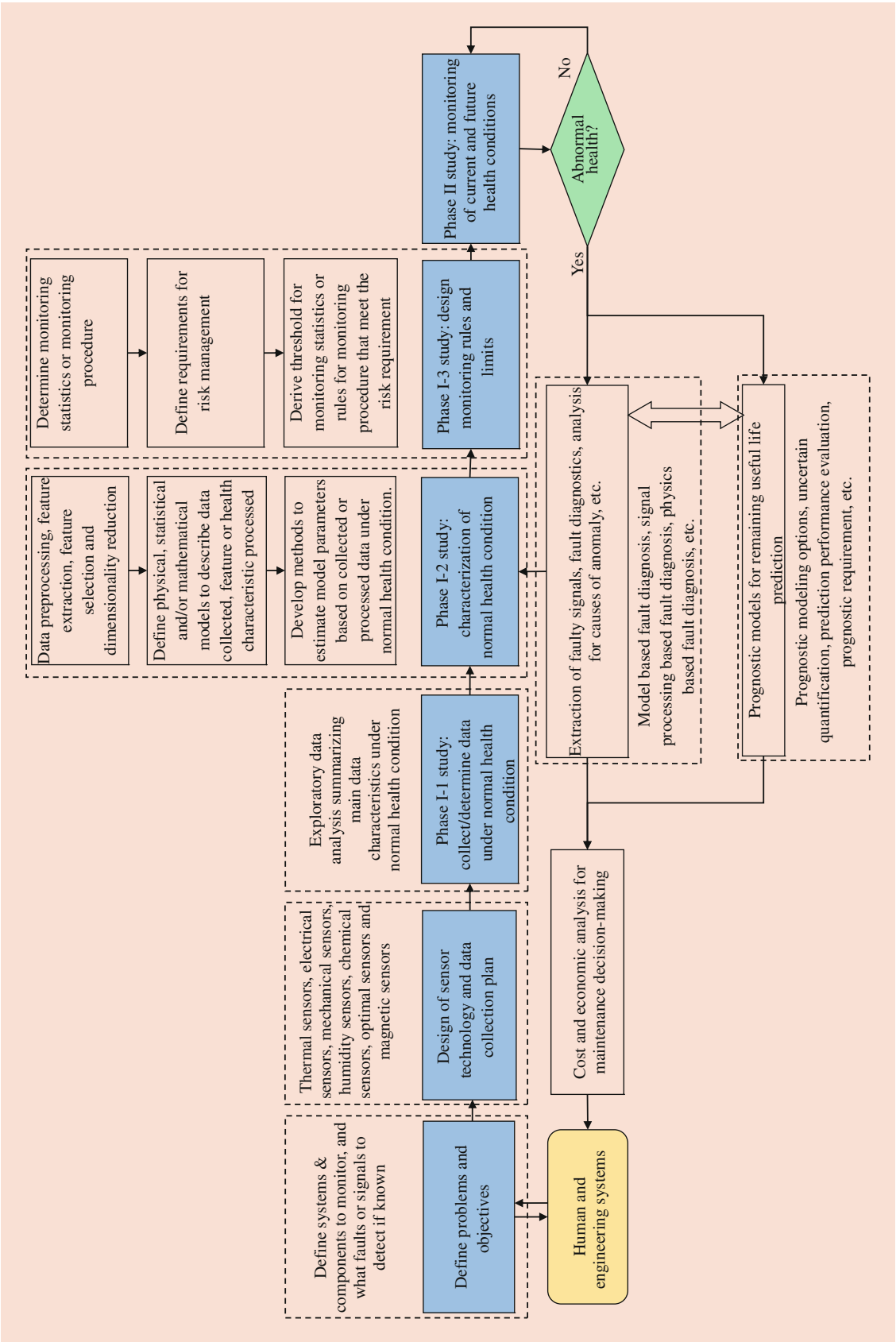


Fig. 11.3 General framework of SHMM

monitoring and surveillance, fault detection and diagnostics, engineering management, supply chain management, and many more. In application to complex human and engineering systems, challenging research problems may arise in various domains driven by big data analytics, such as syndromic surveillance [171, 172], electronics-rich system management [173], simulation and optimization of emergency departments in medical systems [174], and mass transit planning [175].

SHMM covers many broad topics, ranging from experimental design and data collection, data mining and analytics, optimization, decision-making, etc. As more and more systems become data-rich, the theoretical foundation of SHMM is a natural complement to the “data to knowledge to action” paradigm, and therefore to benefit from future developments in data science. Data science shows all the signs of growing into a discipline in its own right, with a strong theoretical foundation at its heart, such foundations being paramount in the development of any new scientific field. Specifically, theoretical research on the foundation of SHMM will build upon theoretical foundational research in data science, which is intrinsically inter-disciplinary. In particular, establishing the theoretical basis of SHMM is likely to involve interdisciplinary collaborations between computer scientists, mathematicians, and statisticians, as these three disciplines are at the heart of the theoretical foundation of SHMM’s closest relative, data science.

For SHMM to make impact to real-life applications, close collaboration is required among SHMM researchers from different disciplines and domain experts. We believe that much of the theoretical foundation of SHMM lies at the intersection of computer science, statistics, and mathematics. Each of those disciplines, however, has been built around particular ideas and in response to particular problems that may have existed for a long time. Thus, the research development of SHMM requires rethinking not only how those three foundational areas interact with each other, but also how each interacts with specific implementations and applications. In particular, the design requirements of business, internet, and social media applications lead to questions that tend to be very different from those in scientific and medical applications in the past. Both the similarities and differences between these areas are striking. Designing the theoretical foundations of SHMM requires paying attention to the problems of researchers implementing SHMM in specific fields as well as to the environments and platforms where computations are to be done. A general framework of SHMM is summarized in Fig. 11.3.

In SHMM, one frequently encounters mixed-type and multi-modality data. For example, a typical dataset may be aggregated from many data sources, including imaging, numerical, graph, text data, etc. Although each specific data type has been researched intensively in isolation, developing

a unified framework will be a more desirable approach to study mixed data systematically. This field has both theoretical and applied implications, and would benefit from a collaboration between statistics, theoretical computer science, mathematics, and practitioners of SHMM. Further research promises to lead to breakthroughs and important progress in science and engineering. A comprehensive review of SHMM can be found in the work by Tsui et al. [176].

References

1. Montgomery, D.C.: Introduction to Statistical Quality Control, 3rd edn. Wiley, New York (1996)
2. Montgomery, D.C., Woodall, W.H.: A discussion on statistically-based process monitoring and control. *J. Qual. Technol.* **29**, 121–162 (1997)
3. Woodall, W.H., Tsui, K.-L., Tucker, G.R.: A review of statistical and fuzzy quality control charts based on categorical data. In: Lenz, H.-J., Wilrich, P. (eds.) *Frontiers in Statistical Quality Control*, vol. 5, pp. 83–89. Physica, Heidelberg (1997)
4. Taguchi, G.: *Introduction to Quality Engineering: Designing Quality into Products and Processes*. Asian Productivity Organization, Tokyo (1986)
5. Phadke, M.S., Kackar, R.N., Speeney, D.V., Grieco, M.J.: Off-line quality control integrated circuit fabrication using experimental design. *Bell Syst. Tech. J.* **1**, 1273–1309 (1983)
6. Kackar, R.N., Shoemaker, A.C.: Robust design: a cost effective method for improving manufacturing process. *ATT Tech. J.* **65**, 39–50 (1986)
7. Woodall, W.H., Montgomery, D.C.: Research issues and ideas in statistical process control. *J. Qual. Technol.* **31**, 376–386 (1999)
8. Lowry, C.A., Montgomery, D.C.: A review of multivariate control charts. *IIE Trans. Qual. Reliab.* **27**, 800–810 (1995)
9. Mason, R.L., Champ, C.W., Tracy, N.D., Wierda, S.J., Young, J.C.: Assessment of multivariate process control techniques. *J. Qual. Technol.* **29**, 140–143 (1997)
10. Kackar, R.N.: Off-line quality control, parameter design, and the Taguchi method. *J. Qual. Technol.* **17**, 176–209 (1985)
11. Leon, R.V., Shoemaker, A.C., Kackar, R.N.: Performance measure independent of adjustment: an explanation and extension of Taguchi’s signal to noise ratio. *Technometrics.* **29**, 253–285 (1987)
12. Box, G.E.P.: Signal to noise ratios, performance criteria and transformations. *Technometrics.* **30**, 1–31 (1988)
13. Nair, V.N.: Taguchi’s parameter design: A panel discussion. *Technometrics.* **34**, 127–161 (1992)
14. Tsui, K.-L.: A critical look at Taguchi’s modelling approach. *J. Appl. Stat.* **23**, 81–95 (1996)
15. Logothetis, N., Haigh, A.: Characterizing and optimizing multi-response processes by the Taguchi method. *Qual. Reliab. Eng. Int.* **4**, 159–169 (1988)
16. Pignatiello, J.J.: Strategies for robust multiresponse quality engineering. *IIE Trans. Qual. Reliab.* **25**, 5–25 (1993)
17. Elsayed, E.A., Chen, A.: Optimal levels of process parameters for products with multiple characteristics. *Int. J. Prod. Res.* **31**, 1117–1132 (1993)
18. Tsui, K.-L.: Robust design optimization for multiple characteristic problems. *Int. J. Prod. Res.* **37**, 433–445 (1999)
19. Shewhart, W.A.: *Economic Control of Quality of Manufactured Product*. Van Nostrand, New York (1931)
20. Box, G.E.P., Kramer, T.: Statistical process monitoring and feedback adjustment - a discussion. *Technometrics.* **34**, 251–285 (1992)

21. Deming, W.E.: *The New Economics: For Industry, Government, Education*, 2nd edn. MIT Center for Advanced Engineering Study, Cambridge (1996)
22. Roberts, S.W.: Control chart tests based on geometric moving averages. *Technometrics*. **1**, 239–250 (1959)
23. Lucas, J.M., Saccucci, M.S.: Exponentially weighted moving average control schemes: properties and enhancements. *Technometrics*. **32**, 1–12 (1990)
24. Page, E.S.: Continuous inspection schemes. *Biometrika*. **41**, 100–115 (1954)
25. Harris, T.J., Ross, W.M.: Statistical process control for correlated observations. *Can. J. Chem. Eng.* **69**, 48–57 (1991)
26. Alwan, L.C.: Effects of autocorrelation on control chart performance. *Commun. Stat. Theory Methods*. **41**, 1025–1049 (1992)
27. Johnson, R.A., Bagshaw, M.: The effect of serial correlation on the performance of CUSUM test. *Technometrics*. **16**, 103–112 (1974)
28. Bagshaw, M., Johnson, R.A.: The effect of serial correlation on the performance of CUSUM test II. *Technometrics*. **17**, 73–80 (1975)
29. Zhang, N.F.: A statistical control chart for stationary process data. *Technometrics*. **40**, 24–38 (1998)
30. Jiang, W., Tsui, K.-L., Woodall, W.H.: A new SPC monitoring method: the ARMA chart. *Technometrics*. **42**, 399–410 (2000)
31. Alwan, L.C., Roberts, H.V.: Time-series modeling for statistical process control. *J. Bus. Econ. Stat.* **6**, 87–95 (1988)
32. Superville, C.R., Adams, B.M.: An evaluation of forecast-based quality control schemes. *Commun. Stat. Simul. Comput.* **23**, 645–661 (1994)
33. Wardell, D.G., Moskowitz, H., Plante, R.D.: Control charts in the presence of data correlation. *Man. Sci.* **38**, 1084–1105 (1992)
34. Wardell, D.G., Moskowitz, H., Plante, R.D.: Run-length distributions of special-cause control charts for correlated observations. *Technometrics*. **36**, 3–17 (1994)
35. Wiel, S.A.V.: Monitoring processes that wander using integrated moving average models. *Technometrics*. **38**, 139–151 (1996)
36. Montgomery, D.C., Mastrangelo, C.M.: Some statistical process control methods for autocorrelated data. *J. Qual. Technol.* **23**, 179–204 (1991)
37. Jiang, W., Wu, H., Tsung, F., Nair, V.N., Tsui, K.-L.: PID-based control charts for process monitoring. *Technometrics*. **44**, 205–214 (2002)
38. Apley, D.W., Shi, J.: The GLRT for statistical process control of autocorrelated processes. *IIE Trans. Qual. Reliab.* **31**, 1123–1134 (1999)
39. Jiang, W.: Multivariate control charts for monitoring autocorrelated processes. *J. Qual. Technol.* **36**, 367–379 (2004)
40. Apley, D.W., Tsung, F.: The autoregressive T^2 chart for monitoring univariate autocorrelated processes. *J. Qual. Technol.* **34**, 80–96 (2002)
41. Runger, G.C., Willemain, T.R.: Model-based and model-free control of autocorrelated processes. *J. Qual. Technol.* **27**, 283–292 (1995)
42. Runger, G.C., Willemain, T.R.: Batch means charts for autocorrelated data. *IIE Trans. Qual. Reliab.* **28**, 483–487 (1996)
43. Bischak, D.P., Kelton, W.D., Pollock, S.M.: Weighted batch means for confidence intervals in steady-state simulations. *Man. Sci.* **39**, 1002–1019 (1993)
44. Alexopoulos, C., Goldsman, D., Tsui, K.-L., Jiang, W.: SPC monitoring and variance estimation. In: Lenz, H.-J., Wilrich, P.T. (eds.) *Frontiers in Statistical Quality Control*, vol. 7, pp. 194–210. Physica, Heidelberg (2004)
45. Box, G.E.P., Jenkins, G.M.: *Time Series Analysis, Forecasting and Control*. Prentice-Hall, Englewood Cliffs (1976)
46. Box, G.E.P., Luceno, A.: *Statistical Control by Monitoring and Feedback Adjustment*. Wiley, New York (1997)
47. Van der Wiel, S.A., Tucker, W.T., Faltin, F.W., Doganaksoy, N.: Algorithmic statistical process control: concepts and application. *Technometrics*. **34**, 278–281 (1992)
48. Tucker, W.T., Faltin, F.W., Van der Wiel, S.A.: Algorithmic statistical process control: an elaboration. *Technometrics*. **35**, 363–375 (1993)
49. Sachs, E., Hu, A., Ingolfsson, A.: Run by run process control: combining SPC and feedback control. *IEEE Trans. Semicond. Manuf.* **8**, 26–43 (1995)
50. Messina, W.S., Montgomery, D.C., Keats, J.B., Runger, G.C.: Strategies for statistical monitoring of integral control for the continuous process industry. In: Keats, J.B., Montgomery, D.C. (eds.) *Statistical Applications in Process Control*, pp. 193–215. Marcel-Dekker, New York (1996)
51. Capilla, C., Ferrer, A., Romero, R., Hualda, A.: Integration of statistical and engineering process control in a continuous polymerization process. *Technometrics*. **41**, 14–28 (1999)
52. Jiang, W., Tsui, K.-L.: An economic model for integrated APC and SPC control charts. *IIE Trans. Qual. Reliab.* **32**, 505–513 (2000)
53. Jiang, W., Tsui, K.-L.: SPC monitoring of MMSE- and PI-controlled processes. *J. Qual. Technol.* **34**, 384–398 (2002)
54. Kourti, T., Nomikos, P., MacGregor, J.F.: Analysis, monitoring and fault diagnosis of batch processes using multiblock and multiway PLS. *J. Process Control*. **5**, 277–284 (1995)
55. Li, W., Wu, C.F.J.: An integrated method of parameter design and tolerance design. *Qual. Eng.* **11**, 417–425 (1999)
56. Tsung, F., Apley, D.W.: The dynamic T-squared chart for monitoring feedback-controlled processes. *IIE Trans. Qual. Reliab.* **34**, 1043–1054 (2002)
57. Jiang, W.: A joint spc monitoring scheme for APC-controlled processes. *IIE Trans. Qual. Reliab.*, 1201–1210 (2004)
58. Boyles, R.A.: Phase I analysis for autocorrelated processes. *J. Qual. Technol.* **32**, 395–409 (2000)
59. Cryer, J.D., Ryan, T.P.: The estimation of Sigma for an X chart: MR/d2 or S/d4 ? *J. Qual. Technol.* **22**, 187–192 (1990)
60. Berube, J., Nair, V.: Exploiting the inherent structure in robust parameter design experiments. *Stat. Sinica*. **8**, 43–66 (1998)
61. Luceno, A.: Performance of discrete feedback adjustment schemes with dead band under stationary versus non-stationary stochastic disturbance. *Technometrics*. **27**, 223–233 (1998)
62. Apley, D.W., Lee, H.C.: Design of exponentially weighted moving average control charts for autocorrelated processes with model uncertainty. *Technometrics*. **45**, 187–198 (2003)
63. Hotelling, H.: Multivariate quality control. In: Eisenhart, C., Hastay, M.W., Wallis, W.A. (eds.) *Techniques of Statistical Analysis*. McGraw-Hill, New York (1947)
64. Lowry, C.A., Woodall, W.H., Champ, C.W., Rigdon, S.E.: A multivariate exponential weighted moving average control chart. *Technometrics*. **34**, 46–53 (1992)
65. Crosier, K.B.: Multivariate generalizations of cumulative sum quality control schemes. *Technometrics*. **30**, 291–303 (1988)
66. Pignatiello, J.J., Runger, G.C.: Comparisons of multivariate CUSUM charts. *J. Qual. Technol.* **22**, 173–186 (1990)
67. Hayter, A.J., Tsui, K.-L.: Identification and quantification in multivariate quality control problems. *J. Qual. Technol.* **26**, 197–208 (1994)
68. Jiang, W., Tsui, K.-L.: Comparison of individual multivariate control charts, submitted for publication
69. Mastrangelo, C.M., Runger, G.C., Montgomery, D.C.: Statistical process monitoring with principal components. *Qual. Reliab. Eng. Int.* **12**, 203–210 (1996)
70. Choi, S.W., Martin, E.B., Morris, A.J., Lee, I.B.: Adaptive multivariate statistical process control for monitoring time-varying processes. *Ind. Eng. Chem. Res.* **45**, 3108–3118 (2006)
71. Li, J., Jin, J.H., Shi, J.J.: Causation-based T-2 decomposition for multivariate process monitoring and diagnosis. *J. Qual. Technol.* **40**, 46–58 (2008)
72. Wang, K.B., Jiang, W.: High-dimensional process monitoring and fault diagnosis via variable selection. *J. Qual. Technol.* **41**(3), 247–258 (2009)

73. Zou, C., Qiu, P.: Multivariate statistical process control using LASSO. *J. Am. Stat. Assoc.* **104**, 1586–1596 (2009)
74. Zou, C., Jiang, W., Tsung, F.: A LASSO-based diagnostic framework for multivariate statistical process control. *Technometrics*. **53**(3), 297–309 (2011)
75. Jiang, W., Wang, K.B., Tsung, F.: A variable-selection-based multivariate EWMA chart for high-dimensional process monitoring and diagnosis. *J. Qual. Technol.* **44**(3), 209–230 (2012)
76. Deng, H., Runger, G., Tuv, E.: System monitoring with real-time contrasts. *J. Qual. Technol.* **44**(1), 9–27 (2012)
77. Wei, Q.M., Huang, W.P., Jiang, W., Li, Y.T.: Projection-based real-time process monitoring and empirical divergence. *IEEE Intell. Syst.* **30**(6), 13–16 (2015)
78. Wei, Q.M., Huang, W.P., *Jiang, W., and Zhao, W.H. (2016) “Real-time process monitoring using kernel distances”, *Int. J. Prod. Res.*, 54(21), 6563–6578
79. He, S.H., *Jiang, W., Deng, H.T. (2018) “A distance-based control chart for monitoring multivariate processes using support vector machines”, *Ann. Oper. Res.*, 263(1), 191–207
80. Woodall, W.H., Spitzner, D.J., Montgomery, D.C., Gupta, S.: Using control charts to monitor process and product quality profiles. *J. Qual. Technol.* **36**(3), 309–320 (2004)
81. Paynabar, K., Zou, C., Qiu, P.: A change-point approach for phase-I analysis in multivariate profile monitoring and diagnosis. *Technometrics*. **58**(2), 191–204 (2016)
82. Woodall, W.H.: Current research on profile monitoring. *Production*. **17**(3), 420–425 (2007)
83. Welch, W.J., Yu, T.-K., Kang, S.M., Sacks, J.: Computer experiments for quality control by parameter design. *J. Qual. Technol.* **22**, 15–22 (1990)
84. Ghosh, S., Derderian, E.: Robust experimental plan and its role in determining robust design against noise factors. *The Statistician*. **42**, 19–28 (1993)
85. Miller, A., Sitter, R.R., Wu, C.F.J., Long, D.: Are large Taguchi-style experiments necessary? A reanalysis of gear and pinion data. *Qual. Eng.* **6**, 21–38 (1993)
86. Lucas, J.M.: Using response surface methodology to achieve a robust process. *Annu. Qual. Congr. Trans.*, Milwaukee, WI. **45**, 383–392 (1990)
87. Rosenbaum, P.R.: Some useful compound dispersion experiments in quality design. *Technometrics*. **38**, 248–260 (1996)
88. Wu, C.F.J., Hamada, M.: *Experiments: Planning, Analysis and Parameter Design Optimization*. Wiley, New York (2000)
89. Kacker, R.N., Tsui, K.-L.: Interaction graphs: graphical aids for planning experiments. *J. Qual. Technol.* **22**, 1–14 (1990)
90. Kacker, R.N., Lagergren, E.S., Filliben, J.J.: Taguchi’s fixed-element arrays are fractional factorials. *J. Qual. Technol.* **23**, 107–116 (1991)
91. Hou, X., Wu, C.F.J.: On the determination of robust settings in parameter design experiments, Univ. Michigan Tech. Rep. 321 (1997)
92. Bingham, D., Sitter, R.: Minimum-aberration two-level fractional factorial split-plot designs. *Technometrics*. **41**, 62–70 (1999)
93. Dumouchel, W., Jones, B.: A simple bayesian modification of D-optimal designs to reduce dependence on an assumed model. *Technometrics*. **36**, 37–47 (1994)
94. Atkinson, A.C., Cook, R.D.: D-optimum designs for heteroscedastic linear models. *J. Am. Stat. Assoc.* **90**, 204–212 (1994)
95. Vining, G.G., Schaub, D.: Experimental designs for estimating both mean and variance functions. *J. Qual. Technol.* **28**, 135–147 (1996)
96. Chang, S.: An algorithm to generate near D-optimal designs for multiple response surface models. *IIE Trans. Qual. Reliab.* **29**, 1073–1081 (1997)
97. Mays, D.P.: Optimal central composite designs in the presence of dispersion effects. *J. Qual. Technol.* **31**, 398–407 (1999)
98. Pledger, M.: Observable uncontrollable factors in parameter design. *J. Qual. Technol.* **28**, 153–162 (1996)
99. Rosenbaum, P.R.: Blocking in compound dispersion experiments. *Technometrics*. **41**, 125–134 (1999)
100. Li, W., Nachtsheim, C.J.: Model-robust factorial designs. *Technometrics*. **42**, 345–352 (2000)
101. Tsui, K.L., Li, A.: Analysis of smaller-and-larger-the-better robust design experiments. *Int. J. Ind. Eng.* **1**, 193–202 (1994)
102. Berube, J., Wu, C.F.J.: Signal-to-noise ratio and related measures parameter design optimization, Univ. Michigan Tech. Rep. 321 (1998)
103. Maghsoodloo, S.: The exact relation of Taguchi’s signal-to-noise ratio to his quality loss function. *J. Qual. Technol.* **22**, 57–67 (1990)
104. Leon, R.V., Wu, C.F.J.: Theory of performance measures in parameter design. *Stat. Sinica*. **2**, 335–358 (1992)
105. Nair, V.N., Pregibon, D.: A data analysis strategy for quality engineering experiments. *ATT Tech. J.* **65**, 73–84 (1986)
106. Vining, G.G., Myers, R.H.: Combining Taguchi and response surface philosophies: a dual response approach. *J. Qual. Technol.* **22**, 38–45 (1990)
107. Copeland, K.A., Nelson, P.R.: Dual response optimization via direct function minimization. *J. Qual. Technol.* **28**, 331–336 (1996)
108. Del Castillo, E., Montgomery, D.C.: A nonlinear programming solution to the dual response problem. *J. Qual. Technol.* **25**, 199–204 (1993)
109. Lin, D.K., Tu, W.: Dual response surface optimization. *J. Qual. Technol.* **28**, 496–498 (1995)
110. Harrington, E.C.: The desirability function. *Ind. Qual. Control*. **21**, 494–498 (1965)
111. Ng, K.K., Tsui, K.-L.: Expressing variability and yield with a focus on the customer. *Qual. Eng.* **5**, 255–267 (1992)
112. Joseph, V.R., Wu, C.F.J.: Operating window experiments: a novel approach to quality improvement. *J. Qual. Technol.* **34**, 345–354 (2002)
113. Joseph, V.R., Wu, C.F.J.: Failure amplification method: an information maximization approach to categorical response optimization. *Technometrics*. **46**, 1–31 (2004)
114. Myers, R.H., Khuri, A.I., Vining, G.: Response surface alternatives to the Taguchi robust parameter design approach. *Am. Stat.* **46**, 131–139 (1992)
115. Myers, R.H., Kim, Y., Griffiths, K.L.: Response surface methods and the use of noise variables. *J. Qual. Technol.* **29**, 429–440 (1997)
116. Box, G.E.P., Meyer, R.D.: Dispersion effects from fractional designs. *Technometrics*. **28**, 19–28 (1986)
117. Lenth, R.V.: Quick and easy analysis of unreplicated factorials. *Technometrics*. **31**, 469–473 (1989)
118. Pan, G.H.: The impact of unidentified location effects on dispersion-effects identification from unreplicated factorial designs. *Technometrics*. **41**, 313–326 (1999)
119. McGrath, R.N., Lin, D.K.: Confounding of location and dispersion effects in unreplicated fractional factorials. *J. Qual. Technol.* **33**, 129–139 (2001)
120. Nelder, J.A., Wedderburn, R.W.: Generalized linear models. *J. Qual. Technol.* **14**, 370–384 (1972)
121. Nelder, J.A., Lee, Y.G.: Generalized linear models for the analysis of Taguchi type experiments. *J. Qual. Technol.* **7**, 107–120 (1991)
122. Engel, J., Huele, A.F.: Joint modeling of mean and dispersion. *J. Qual. Technol.* **38**, 365–373 (1996)
123. Hamada, M., Nelder, J.A.: Generalized linear models for quality-improvement experiments. *J. Qual. Technol.* **29**, 292–304 (1997)

124. Chipman, H., Hamada, M.: Bayesian analysis of ordered categorical data from industrial experiments. *Technometrics*. **38**, 1–10 (1996)
125. Chipman, H.: Handling uncertainty in analysis of robust design experiments. *J. Qual. Technol.* **30**, 11–17 (1998)
126. Box, G.E.P., Meyer, R.D.: Finding the active factors in fractionated screening experiments. *J. Qual. Technol.* **25**, 94–105 (1993)
127. Ribeiro, J.L., Elsayed, E.A.: A case study on process optimization using the gradient loss function. *Int. J. Prod. Res.* **33**, 3233–3248 (1995)
128. Ribeiro, J.L., Fogliatto, F., ten Caten, C.S.: Minimizing manufacturing and quality costs in multiresponse optimization. *Qual. Eng.* **13**, 191–201 (2000)
129. Kumar, P., Barua, P.B., Gaindhar, J.L.: Quality optimization (multi-characteristics) through Taguchi's technique and utility concept. *Qual. Reliab. Eng. Int.* **16**, 475–485 (2000)
130. Artes-Leon, N.: A pragmatic approach to multiple-response problems using loss functions. *Qual. Eng.* **9**, 475–485 (1996)
131. Ames, A.E., Mattucci, N., MacDonald, S., Szonyi, G., Hawkins, D.M.: Quality loss functions for optimization across multiple response surfaces. *J. Qual. Technol.* **29**, 339–346 (1997)
132. Joseph, V.R.: Quality loss functions for nonnegative variables and their applications. *J. Qual. Technol.* **36**, 129–138 (2004)
133. Derringer, G., Suich, R.: Simultaneous optimization of several response variables. *J. Qual. Technol.* **12**, 214–219 (1980)
134. Kim, K.-J., Lin, D.K.J.: Simultaneous optimization of mechanical properties of steel by maximizing exponential desirability functions. *Appl. Stat.* **49**, 311–325 (2000)
135. Duffy, J., Liu, S.Q., Moskowitz, H., Plante, R., Preckel, P.V.: Assessing multivariate process/product yield via discrete point approximation. *IIE Trans. Qual. Reliab.* **30**, 535–543 (1998)
136. Fogliatto, F.S., Albin, S.L.: Variance of predicted response as an optimization criterion in multiresponse experiments. *Qual. Eng.* **12**, 523–533 (2000)
137. Plante, R.D.: Process capability: a criterion for optimizing multiple response product and process design. *IIE Trans. Qual. Reliab.* **33**, 497–509 (2001)
138. Ghosh, S., Derderian, E.: Determination of robust design against noise factors and in presence of signal factors. *Commun. Stat. Sim. Comp.* **24**, 309–326 (1995)
139. Wasserman, G.S.: Parameter design with dynamic characteristics: a regression perspective. *Qual. Reliab. Eng. Int.* **12**, 113–117 (1996)
140. Miller, A., Wu, C.F.J.: Parameter design for signal-response systems: a different look at Taguchi's dynamic parameter design. *Stat. Sci.* **11**, 122–136 (1996)
141. Lunani, M., Nair, V.N., Wasserman, G.S.: Graphical methods for robust design with dynamic characteristics. *J. Qual. Technol.* **29**, 327–338 (1997)
142. McCaskey, S.D., Tsui, K.-L.: Analysis of dynamic robust design experiments. *Int. J. Prod. Res.* **35**, 1561–1574 (1997)
143. Tsui, K.-L.: Modeling and analysis of dynamic robust design experiments. *IIE Trans. Qual. Reliab.* **31**, 1113–1122 (1999)
144. Joseph, V.R., Wu, C.F.J.: Robust parameter design of multiple target systems. *Technometrics*. **44**, 338–346 (2002)
145. Joseph, V.R., Wu, C.F.J.: Performance measures in dynamic parameter design. *J. Jpn. Qual. Eng. Soc.* **10**, 82–86 (2002)
146. Joseph, V.R.: Robust parameter design with feed-forward control. *Technometrics*. **45**, 284–292 (2003)
147. Mesenbrink, P., Lu, J.C., McKenzie, R., Taheri, J.: Characterization and optimization of a wave-soldering process. *J. Am. Stat. Assoc.* **89**, 1209–1217 (1994)
148. Lin, S.M., Wen, T.C.: Experimental strategy -application of Taguch's quality engineering method to zinc phosphate coating uniformity. *Plat. Surf. Finishing.* **81**, 59–64 (1994)
149. Chhajed, D., Lowe, T.J.: Tooling choices and product performance. *IIE Trans. Qual. Reliab.* **32**, 49–57 (2000)
150. Hamada, M.: Reliability improvement via Taguchi's robust design. *Qual. Reliab. Eng. Int.* **9**, 7–13 (1993)
151. Kuhn, A.M., Carter, W.H., Myers, R.H.: Incorporating noise factors into experiments with censored data. *Technometrics*. **42**, 376–383 (2000)
152. D'errico, J.R., Zaino, N.A.: Statistical tolerancing using a modification of Taguchi's method. *Technometrics*. **30**, 397–405 (1988)
153. Bisgaard, S.: Designing experiments for tolerancing assembled products. *Technometrics*. **39**, 142–152 (1997)
154. Zhang, C.C., Wang, H.P.: Robust design of assembly and machining tolerance allocations. *IIE Trans. Qual. Reliab.* **30**, 17–29 (1998)
155. Maghsoodloo, S., Li, M.H.: Optimal asymmetric tolerance design. *IIE Trans. Qual. Reliab.* **32**, 1127–1137 (2000)
156. Moskowitz, H., Plante, R., Duffy, J.: Multivariate tolerance design using quality loss. *IIE Trans. Qual. Reliab.* **33**, 437–448 (2001)
157. Bowles, J.B.: A survey of reliability-prediction procedures for microelectronic devices. *IEEE Trans. Reliab.* **41**(1), 2–12 (1992)
158. Tsui, K.-L., Chiu, W., Gierlich, P., Goldsman, D., Liu, X., Maschek, T.: A review of healthcare, public health, and syndromic surveillance. *Qual. Eng.* **20**(4), 435–450 (2008)
159. Pecht, M.: *Prognostics and Health Management of Electronics*. Wiley, Hoboken (2008)
160. Di Maio, F., Hu, J., Tse, P., Pecht, M., Tsui, K., Zio, E.: Ensemble approaches for clustering health status of oil sand pumps. *Expert Syst. Appl.* **39**(5), 4847–4859 (2012)
161. Bellcore: Reliability prediction procedure for electronic equipment. *Tech. Rep. TR-NWT-000332*, 83–86 (1990)
162. Pecht, M., Ramappan, V.: Are components still the major problem. A review of electronic system and device field failure returns. *IEEE Trans. Components, Hybrids Manuf. Technol.* **15**(6), 1160–1164 (1992)
163. Thomas, D.A., Avers, K., Pecht, M.: The "troublenotidentified" phenomenon in automotive electronics. *Microelectron. Reliab.* **42**(4–5), 641–651 (2002)
164. Qi, H., Ganesan, S., Pecht, M.: No-fault-found and intermittent failures in electronic products. *Microelectron. Reliab.* **48**(5), 663–674 (2008)
165. Williams, R., Banner, J., Knowles, I., Dube, M., Natishan, M., Pecht, M.: An investigation of 'cannot duplicate' failures. *Qual. Reliab. Eng. Int.* **14**(5), 331–337 (1998)
166. Anderson, K.: Intermittent fault detection & isolation system. In: *Proceedings of the Maintenance Symposium and Exhibition, Department of Defense, November 2012*
167. Keane, A.G.: *Toyota Finds Most Sudden Acceleration Crashes Are Driver Error*. Bloomberg, Washington, DC (2010)
168. Plett, G.L.: Sigma-point Kalman filtering for battery management systems of LiPB-based HEV battery packs: part 2: simultaneous state and parameter estimation. *J. Power Sources*. **161**, 1369–1384 (2006)
169. Tsui, K.-L., Chiu, W., Gierlich, P., Goldsman, D., Liu, X., Maschek, T.: A review of healthcare, public health, and syndromic surveillance. *Qual. Eng.* **20**, 435–450 (2008)
170. Tsui, K.L., Chen, N., Zhou, Q., Hai, Y., Wang, W.: Prognostics and health management: a review on data driven approaches. *Math. Probl. Eng.* **2015** (2015)
171. Ginsberg, J., Mohebbi, M.H., Patel, R.S., Brammer, L., Smolinski, M.S., Brilliant, L.: Detecting influenza epidemics using search engine query data. *Nature*. **457**, 1012–1014 (2009)
172. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Byers, A.H.: *Big data: the next frontier for innovation, competition, and productivity* (2011)

173. Pecht, M., Jaai, R.: A prognostics and health management roadmap for information and electronics-rich systems. *Microelectron. Reliab.* **50**, 317–323 (2010)
174. Guo, H., Goldman, D., Tsui, K.-L., Zhou, Y., Wong, S.-Y.: Using simulation and optimisation to characterise durations of emergency department service times with incomplete data. *Int. J. Prod. Res.* **54**, 6494–6511 (2016)
175. Wang, W., Lo, S., Liu, S.: Aggregated metro trip patterns in urban areas of Hong Kong: evidence from automatic fare collection records. *Int. J. Urban Plan. Dev.* **141**, 05014018 (2014)
176. Tsui, K.-L., Zhao, Y., Wang, D.: Big data opportunities: system health monitoring and management. *IEEE Access.* **7**, 68853–68867 (2019)



Wei Jiang is a distinguished professor in Antai College of Economics and Management at Shanghai Jiao Tong University. He obtained his Ph.D. degree in industrial engineering and engineering management from Hong Kong University of Science and Technology in 2000. His current research activities include statistical methods for quality control, machine learning, and enterprise intelligence.



Terrence E. Murphy is an associate professor at the Yale University School of Medicine and Director of Biostatistics of the Claude D. Pepper Older Americans Independence Center at Yale (P30AG021342). He earned his Ph.D. in industrial and systems engineering from the Georgia Institute of Technology in 2004. Prior to his graduate work in engineering statistics, he worked for the Eastman Kodak and Johnson & Johnson companies in the manufacture and development of clinical instrumentation. His interests include Bayesian statistics and the development and validation of risk prediction models.



Kwok-Leung Tsui is a chaired professor in the School of Data Science at City University of Hong Kong, and the founder and Director of Center for Systems Informatics Engineering. Prior to joining City University of Hong Kong, Professor Tsui was Professor at the School of Industrial and Systems Engineering at the Georgia Institute of Technology. Professor Tsui was a recipient of the National Science Foundation Young Investigator Award. He is Fellow of the American Statistical Association, American Society for Quality, and International Society of Engineering Asset Management; a US representative to the ISO Technical Committee on Statistical Methods. Professor Tsui was Chair of the INFORMS Section on Quality, Statistics, and Reliability and the Founding Chair of the INFORMS Section on Data Mining. Professor Tsui's current research interests include data mining, surveillance in healthcare and public health, prognostics and systems health management, calibration and validation of computer models, process control and monitoring, and robust design and Taguchi methods.



Yang Zhao is a scientific officer in the Centre for Systems Informatics Engineering at City University of Hong Kong. She received her Bachelor's degree in Statistics from Shandong University of Science and Technology in 2011 and her Ph.D. degree from City University of Hong Kong in 2015. Her research interests are in machine learning and statistics, especially their application to real life applications.



Chain Sampling

Govindaraju Kondaswamy

Contents

12.1	Introduction	221
12.2	ChSP-1 Chain Sampling Plan	222
12.3	Extended Chain Sampling Plans	223
12.4	Two-Stage Chain Sampling	224
12.5	Modified ChSP-1 Plan	225
12.6	Chain Sampling and Deferred Sentencing	227
12.7	Comparison of Chain Sampling with Switching Systems	229
12.8	Chain Sampling for Variable Inspection	230
12.9	Chain Sampling and CUSUM	232
12.10	Other Interesting Extensions	233
12.11	Concluding Remarks	233
	References	234

sixth section of this chapter is on the relationship between chain/dependent sampling and deferred sentencing type of plans. A review of sampling inspection plans that are based on the ideas of chain or dependent sampling or deferred sentencing is also made in this section. A large number of recent publications based on the idea of chaining past and future lot results are also reviewed. The economics of chain sampling when compared to the two-plan quick switching system is discussed in the seventh section. The eighth section extends the attribute chain sampling rule to variables inspection. In the ninth section, chain sampling is compared with the well-known CUSUM approach for attribute data. The tenth section gives several other interesting extensions such as chain sampling for mixed inspection and process control. The final section gives the concluding remarks.

Abstract

A brief introduction to the concept of chain sampling for quality inspection is first presented. The chain sampling plan of type ChSP-1 selectively chains the past inspection results. A discussion on the design and application of ChSP-1 plans is presented in the second section of this chapter. Various extensions of chain sampling plans such as ChSP-4 plan are discussed in the third part. Representation of the ChSP-1 plan as a two-stage cumulative results criterion plan and its design are discussed in the fourth part. The fifth section relates to the modification of ChSP-1 plan which results in sampling economy. The

Keywords

Acceptance quality limit · Attribute inspection · Chain sampling · Cumulative result criterion · Deferred sentencing · Destructive testing · Operating characteristic · Quick switching · Sampling economy · Variables inspection

12.1 Introduction

Acceptance sampling is the methodology that deals with procedures by which decision to accept or not accept lots of items is based on the results of the inspection of samples. Special purpose acceptance sampling inspection plans (shortly special purpose plans) are tailored for special applications as against general or universal use. Prof. Harold F. Dodge, who is regarded as the father of acceptance sampling, introduced the idea of chain sampling in [1]. Chain sampling inspection can be viewed as a protocol or plan based on a cumulative

G. Kondaswamy (✉)
 School of Mathematical and Computational Sciences, Massey
 University, Palmerston North, New Zealand
 e-mail: k.govindaraju@massey.ac.nz
<https://www.massey.ac.nz/massey/expertise/profile.cfm?stref=582330>

results criterion (CRC), where related batch information is chained or cumulated. The phrase *chain sampling* is also used in sample surveys to imply snowball sampling for collection of data. It should be noted that this phrase was originally coined in the acceptance sampling literature and should be distinguished from its usage in other areas.

Chain sampling is extended to two or more stages of cumulation of inspection results with appropriate acceptance criteria for each stage. The theory of chain sampling is also closely related to the various other methods of sampling inspection such as dependent-deferred sentencing, tightened-normal-tightened sampling, quick switching inspection, etc. In this chapter, we provide an introduction to chain sampling and discuss briefly various generalizations of chain sampling plans. We also review a few sampling plans which are related to or based on the methodology of chain sampling. The selection or design of various chain sampling plans is also briefly presented.

12.2 ChSP-1 Chain Sampling Plan

A single sampling attributes inspection plan calls for acceptance of a lot under consideration if the number of nonconforming units found in a random sample of size n is less than or equal to the acceptance number Ac . Whenever the operating characteristic (OC) curve of a single sampling plan is required to pass through a prescribed point, the sample size n will be an increasing function of the acceptance number Ac . This fact can be verified from the table of np or *unity* values given in [2] for various values of the probability of acceptance $P_a(p)$ of the lot under consideration whose fraction nonconforming units is p . The same result is true when the OC curve has to pass through two predetermined points, usually one at the top and the other at the bottom of the OC curve (see [3]). Thus, for situations where small sample sizes are preferred, only single sampling plans with $Ac = 0$ are desirable (see [4]). However, as observed by Dodge [1] and several authors, the $Ac = 0$ plan has a “pathological” OC curve in that the curve starts to drop rapidly even for a very small increase in the proportion or fraction nonconforming. In other words, the OC curve of the $Ac = 0$ plan has no point of inflection. Whenever a sampling plan for costly or destructive testing is required, it is common to force the OC curve to pass through a point, say (LQL, β), where LQL is the limiting quality level for ensuring consumer protection and β is the associated consumer’s risk. All other sampling plans such as double and multiple sampling plans will require more sample size for a one-point protection such as (LQL, β). Unfortunately, the $Ac = 0$ plan has the following two disadvantages:

1. The OC curve of the $Ac = 0$ plan has no point of inflection and hence it starts to drop rapidly even for a smallest increase in the fraction nonconforming p .
2. The producer dislikes an $Ac = 0$ plan because a single occasional nonconformity will call for the rejection of the lot.

The chain sampling plan ChSP-1 of [1] is an answer to the question whether anything can be done to improve the “pathological” shape of the OC curve of a zero acceptance number plan. A production process, when in a state of statistical control, maintains nearly a constant but unknown fraction nonconforming p . If a series of lots formed from such a stable process is submitted for inspection, known as a Type B situation, then the samples drawn from the submitted lots are simply random samples drawn directly from the production process. Hence, it is logical to allow a single occasional nonconforming unit in the current sample whenever the evidence of good past quality, as demonstrated by the i preceding samples containing no nonconforming units, is available. Alternatively, we can *chain* or *cumulate* the results of past lot inspections to take a decision on the current lot without increasing the sample size.

The operating procedure of the chain sampling plan of type ChSP-1 is formally stated below:

1. From each of the lots submitted, draw a random sample of size n and observe the number of nonconforming units d .
2. Accept the lot if d is zero. Reject the lot if $d > 1$. If $d = 1$, the lot is accepted provided all the samples of size n each drawn from the preceding i lots are free from nonconforming units; otherwise, reject the lot.

Thus the ChSP-1 plan has two parameters, namely, the sample size n and i , the number of preceding sample results chained for making a decision on the current lot. It is also required that the consumer has confidence in the producer, and the producer will not deliberately pass a poor-quality lot taking advantage of the small samples used and the utilization of preceding samples for taking a decision on the current lot.

The ChSP-1 plan always accepts the lot if $d = 0$ and conditionally accepts if $d = 1$. The probability of preceding i samples of size n to be free from nonconforming units is $P_{0,n}^i$. Hence, the OC function is $P_a(p) = P_{0,n} + P_{1,n}P_{0,n}^i$ where $P_{d,n}$ is the probability of getting d nonconforming units in a sample of size n . Figure 12.1 shows the improvement in the shape of the OC curve of the zero acceptance number single sampling plan by the use of chain sampling. Clark [5] provided a discussion on the OC curves of chain sampling plans, a modification, and some applications. Lieberman and Hawley [6] argued in favor of chain sampling because the attribute international sampling standards suffer from

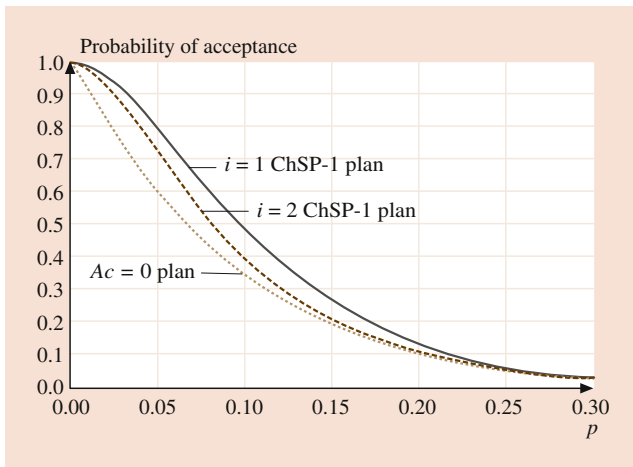


Fig. 12.1 Comparison of OC curves of $Ac = 0$ and ChSP-1 plans

the small or fractional acceptance numbers. Liebesman and Hawley [6] also provided necessary tables and examples for the chain sampling procedures. Most textbooks on statistical quality control also contain a section of chain sampling and provide some applications.

Soundararajan [7, 8] constructed tables for the selection of chain sampling plans for given acceptable quality level (AQL, denoted as p_1), producer’s risk α , LQL (denoted as p_2), and β . ChSP-1 plans found from this source are approximate, and a more accurate procedure that also minimizes the sum of actual producer’s and consumer’s risks is given in [9]. Table 12.1, adapted from [9], is based on the binomial distribution for OC curve of the ChSP-1 plan. This table can also be used to select ChSP-1 plans for given LQL and β which may be used in place of zero acceptance number plans.

Ohta [10] investigated the performance of ChSP-1 plans using the Graphical Evaluation and Review Technique (GERT) and derived measures such as OC, average sample number (ASN) for the ChSP-1 plan. Raju and Jothikumar [11] provided a ChSP-1 plan design procedure based on Kullback-Leibler information and necessary tables for the selection of the plan. Govindaraju [12] discussed the design ChSP-1 plan for minimum average total inspection (ATI). There are a number of other sources where the ChSP-1 plan design is discussed. This chapter provides additional references on designing chain sampling plans, inter alia, discussing various extensions and generalizations.

12.3 Extended Chain Sampling Plans

Frishman [13] extended the ChSP-1 plan and developed ChSP-4 and ChSP-4A plans which incorporate a rejection number greater than 1. Both ChSP-4 and ChSP-4A plans are

Table 12.1 ChSP-1 plans indexed by AQL and LQL ($\alpha = 0.05$, $\beta = 0.10$) for fraction nonconforming inspection

AQL in %	LQL in %					
	0.1	0.15	0.25	0.4	0.65	1
1.5	154:02					
2.0	114:04	124:01				
2.5	91:04	92:02				
3.0	76:03	76:03	82:01			
3.5	65:03	65:03	70:01			
4.0	57:02	57:02	57:02			
4.5	51:02	51:02	51:02		Key $n : i$	
5.0	45:03	45:03	45:03	49:01		
5.5	41:03	41:03	41:03	45:01		
6.0	38:03	38:02	38:02	38:02		
6.5	35:03	35:02	35:02	35:02		
7.0	32:03	32:03	32:03	32:03		
7.5	30:03	30:03	30:02	30:02		
8.0	28:03	28:03	28:02	28:02	30:01	
8.5	26:03	26:03	26:03	26:03	29:01	
9.0	25:03	25:03	25:02	25:02	27:01	
9.5	24:03	24:03	24:02	24:02	24:02	
10	22:03	22:03	22:03	22:03	23:02	
11	20:03	20:03	20:02	20:02	20:02	
12	19:03	19:03	19:02	19:02	19:02	20:01
13	17:03	17:03	17:03	17:02	17:02	18:01
14	16:03	16:03	16:03	16:02	16:02	16:02
15	15:03	15:03	15:03	15:02	15:02	15:02

Table 12.2 ChSP-4A plan

Stage	Sample size	Acceptance number	Rejection number
1	n	a	r
2	$(k - 1)n$	b	$b + 1$

operated like a traditional double sampling attributes plan but use $(k - 1)$ past lot results instead of actually taking a second sample from the current lot. Table 12.2 is a compact tabular representation of Frishman’s ChSP-4A plan.

ChSP-4 plan restricts r to $b + 1$ which means that the same rejection number is used for both stages. Conditional double sampling plans of [14] and the partial and full link sampling plans of [15] are actually particular cases of the ChSP-4A plan when $k = 2$ and $k = 3$, respectively. However the fact that the OC curves of these plans are the same as the ChSP-4A plan is not reported in both papers (see [16]).

Extensive tables for the selection of ChSP-4 and ChSP-4A plans under various selection criteria were constructed by Raju [17, 18], Raju and Murthy [19, 20], and Raju and Jothikumar [21]. Raju and Jothikumar [21] provided a complete summary of various selection procedures for ChSP-4 and ChSP-4A plans and also discussed two further types of

optimal plans: the first one involving minimum risks and the second one based on Kullback-Leibler information. Unfortunately, the tables given in [17–21] for the ChSP-4 or ChSP-4A design require the user to specify the acceptance and rejection numbers. This serious design limitation is not an issue with the procedures and computer programs developed by Vaerst [22] who discussed the design of ChSP-4A plans involving minimum sample sizes for given AQL, α , LQL, and β without assuming any specific acceptance numbers. Raju [17, 18], Raju and Murthy [19, 20], and Raju and Jothikumar [21] considered a variety of design criteria, while [22] discussed only the (AQL, LQL) criterion. The ChSP-4 and ChSP-4A plans were obtained from [17–21]. The procedure given in [21] can be used in any Type B situation of a series of lots from a stable production process, not necessarily when the product involved costly or destructive testing. This is because the acceptance numbers covered are well over zero. The major disadvantage of [13] extended ChSP-4 and ChSP-4A plans is that the neighboring lot information is not always utilized. Vidya [23] considered the variance of the outgoing quality limit (VOQL) criterion for designing ChSP-4 plans. Even though ChSP-4 and ChSP-4A plans require smaller sample size than the traditional double sampling plans, these plans may not be economical to other comparable conditional sampling plans.

Bagchi [24] presented an extension of the ChSP-1 plan, which calls for additional sampling only when one nonconforming unit is found. The operating procedure of Bagchi's extended chain sampling plan is given below:

1. At the outset, inspect n_1 units selected randomly from each lot. Accept the lot if all the n_1 units are conforming; otherwise, reject the lot.
2. If i successive lots are accepted, then inspect only $n_2 < n_1$ items from each of the submitted lots. Accept the lot as long as no nonconforming units are found. If two or more nonconforming units are found, reject the lot. In the event of one nonconforming unit is found in n_2 inspected units, then inspect a further sample $(n_1 - n_2)$ units from the same lot. Accept the lot under consideration if no further nonconforming units are found in the additional $(n_1 - n_2)$ inspected units; otherwise, reject the lot.

Representing Bagchi's plan as a Markov chain, [25] derived the steady-state OC function and few other performance measures.

Gao [26] considered the effect of inspection errors on a chain sampling plan with two acceptance numbers and also provided design procedures. Application of chain sampling for a reliability acceptance test for exponential life times is also given in [26].

12.4 Two-Stage Chain Sampling

Dodge and Stephens [27] viewed the chain sampling approach as a cumulative result criterion (CRC) applied in two stages and extended it to include larger acceptance numbers. Their approach calls for the first stage of cumulation of a maximum of k_1 consecutive lot results, during which acceptance is allowed if the maximum allowable nonconforming units is c_1 or less. After passing the first stage of cumulation (i.e., when consecutive lots are accepted), the second stage cumulation of $k_2 (> k_1)$ lot results begins. In the second stage of cumulation, an acceptance number of $c_2 (> c_1)$ is applied. Stephens and Dodge [28] presented a further generalization of the family of "two-stage" chain sampling inspection plans by using different sample sizes in the two stages. The complete operating procedure of a generalized two-stage chain sampling plan is stated below:

1. At the outset, draw a random sample of n_1 units from the first lot. In general, a sample of size n_j ($j = 1, 2$) will be taken while operating in j th stage of cumulation.
2. Record d the number of nonconforming units in each sample, as well as D the cumulative number of nonconforming units from the first and up to, and including, the current sample. As long as $D_i \leq c_1$ ($1 \leq i \leq k_1$), accept the i th lot.
3. If k_1 consecutive lots are accepted, continue to cumulate the number of nonconforming units D in the k_1 samples plus additional samples up to but no more than k_2 samples. During this second stage of cumulation, accept the lots as long as $D_i \leq c_2$ ($k_1 < i \leq k_2$).
4. After passing the second stage of k_2 lot acceptances, start cumulation as a moving total over k_2 consecutive samples (by adding the current lot result and dropping the preceding k_2 th lot result). Continue to accept lots as long as $D_i \leq c_2$ ($i > k_2$).
5. In any stage of sampling, reject the lot if $D_i > c_i$ and return to Step 1 (a fresh restart of the cumulation procedure).

Figure 12.2 shows how the cumulative result criterion is used in a two-stage chain sampling plan for $k_1 = 3$ and $k_2 = 5$.

An important subset of the generalized two-stage chain sampling plan is when $n_1 = n_2$, and this subset is designated as ChSP- (c_1, c_2) , which has five parameters n, k_1, k_2, c_1 , and c_2 . The original chain sampling plan ChSP-1 of [1] is a further subset of the ChSP- $(0,1)$ plan with $k_1 = k_2 - 1$, that is, the OC curve of the generalized two-stage chain sampling plan is equivalent to the OC curve of the ChSP-1 plan when $k_1 = k_2 - 1$. Dodge and Stephens [27] derived the OC function of ChSP- $(0,1)$ plan as

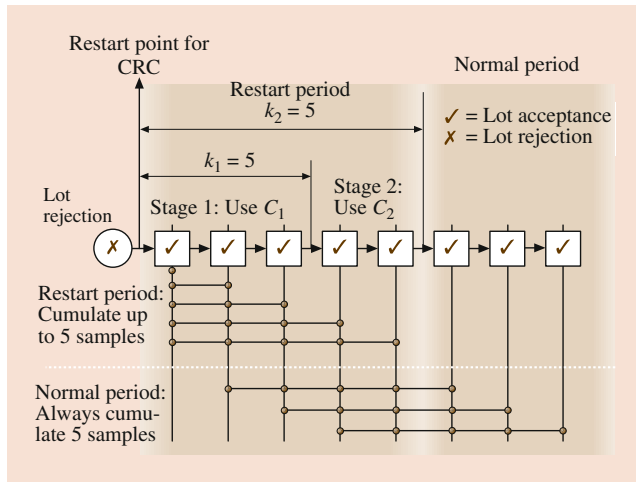


Fig. 12.2 Operation of a two-stage chain sampling plan with $k_1 = 3$ and $k_2 = 5$

$$P_a(p) = \frac{P_{0,n}(1 - P_{0,n}) + P_{0,n}^{k_1} P_{1,n} (1 - P_{0,n}^{k_2 - k_1})}{1 - P_{0,n} + P_{0,n}^{k_1} P_{1,n} (1 - P_{0,n}^{k_2 - k_1})}, \quad k_2 > k_1.$$

Both the ChSP-1 and ChSP-(0,1) plans overcome the disadvantages of the zero acceptance number plan mentioned earlier. The operating procedure of the ChSP-(0,1) plan can be succinctly stated as follows:

1. A random sample of size n is taken from each successive lot, and the number of nonconforming units in each sample is recorded, as well as the cumulative number of nonconforming units found so far.
2. Accept the lot associated with each new sample as long as no nonconforming units are found.
3. Once k_1 lots have been accepted, accept subsequent lots as long as the cumulative number of nonconforming units is no greater than one.
4. Once $k_2 > k_1$ lots have been accepted, cumulate the number of nonconforming units over the most k_2 lots, and continue to accept as long as this cumulative number of nonconforming units is one or none.
5. If, at any stage, the cumulative number of nonconforming units becomes greater than one, reject the current lot and return to Step 1.

Procedures and tables for the design of ChSP-(0,1) plan are available in [29, 30]. Govindaraju and Subramani [31] showed that the choice of $k_1 = k_2 - 1$ is always forced on the parameters of the ChSP-(0,1) plan when a plan is selected for given AQL, α LQL, and β , that is, a ChSP-1 plan will be sufficient, and one need not opt for a two-stage cumulation of nonconforming units.

In various technical reports of the Statistics Center, Rutgers University (see [32] for a list), Stephens and Dodge formulated the two-stage chain sampling plan as a Markov chain and evaluated its performance. The performance measures considered by them include the steady-state OC function, ASN, average run length (ARL), etc. For comparison of chain sampling plans with the traditional or noncumulative plans, two types of ARLs are used. The first type of ARL, i.e., ARL_1 , is the average number of samples to the first rejection after a sudden shift in the process level, say from p_0 to p_s . The usual ARL, i.e., ARL_2 , is the average number of samples to the first rejection given the stable process level p_0 . The difference $(ARL_1 - ARL_2)$ measures the extra lag due to chain sampling. However, this extra lag may be compensated by the gains in sampling efficiency as explained in [33].

Stephens and Dodge [34] summarized the Markov chain approach to evaluate the performance of some selected two-stage chain sampling plans, while more detailed derivations were published in their technical reports. Based on the expressions for the OC function derived by Stephens and Dodge in their various technical reports listed in [32]. Raju and Murthy [35] and Jothikumar and Raju [36] discussed various design procedures for the ChSP-(0,2) and ChSP-(1,2) plans. Raju [37] extended the two-stage chain sampling to three stages and evaluated the OC performances of few selected chain sampling plans fixing the acceptance numbers for the three stages. The three-stage cumulation procedure becomes very complex and will play only a limited role for costly or destructive inspections. The three-stage plan will however be useful for general Type B lot by lot inspections.

12.5 Modified ChSP-1 Plan

In [1], chaining of past lot results does not always occur. It occurs only when a nonconforming unit is observed in the current sample. This means that the available historical evidence of quality is not utilized fully. Govindaraju and Lai [38] developed a modified chain sampling plan (MChSP-1) that always utilizes the recently available lot quality history. The operating procedure of the MChSP-1 plan is given below.

1. From each of the submitted lots, draw a random sample of size n . Reject the lot if one or more nonconforming units are found in the sample.
2. Accept the lot if no nonconforming units are found in the sample provided the preceding i samples also contained no nonconforming units except in one sample which may contain at most one nonconforming unit. Otherwise, reject the lot.

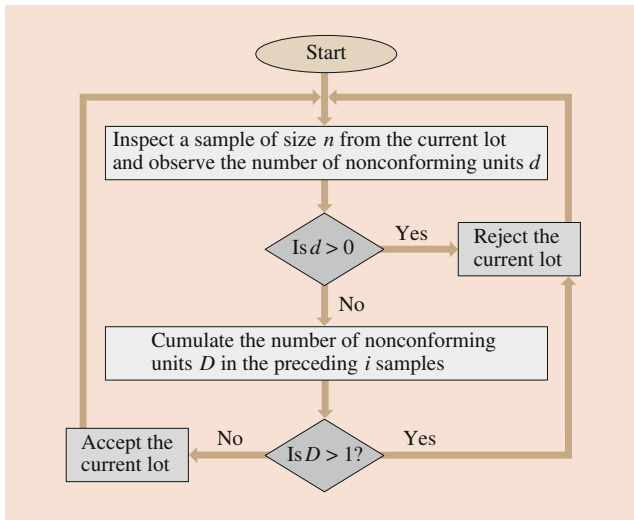


Fig. 12.3 Operation of the MChSP-1 plan

A flow chart showing the operation of the MChSP-1 plan is in Fig. 12.3. MChSP-1 plan allows a single nonconforming unit in any one of the preceding i samples, but the lot under consideration is rejected if the current sample has a nonconforming unit. Thus, the plan gives a “psychological” protection to the consumer in that it allows acceptance only when all the current sample units are conforming. Allowing one nonconforming unit in any one of the preceding i samples is essential to offer protection to the producer, i.e., to achieve an OC curve with a point of inflection. In MChSP-1 plan, rejection of lots would occur until the sequence of submissions advances to a stage where two or more nonconforming units were no longer included in the sequence of i samples. In other words, if two or more nonconforming units are found in a single sample, it will result in i subsequent lot rejections. In acceptance sampling, one has to look at the OC curve to have an idea of the protection to the producer as well as to the consumer, and what happens in an individual sample or for a few lots is not very important. If two or more nonconforming units are found in a single sample, it does not mean that the subsequent lots need not be inspected since they will be automatically rejected under the proposed plan. It should be noted that results of subsequent lots will be utilized continuously and the producer has to show an improvement in quality with one or none nonconforming unit in the subsequent samples in order to permit future acceptances. This will act as a strong motivating factor for quality improvement.

The OC function $P_a(p)$ of the MChSP-1 plan is derived by Govindaraju and Lai [38] as $P_a(p) = P_{0,n}(P_{0,n}^i + iP_{0,n}^{i-1}P_{1,n})$. Figure 12.4 compares the OC curves of ChSP-1 and MChSP-1 plans. From Fig. 12.4, we observe that the MChSP-1 plan decreases the probability of acceptance at poor-quality levels but maintains the probability of acceptance at good-quality

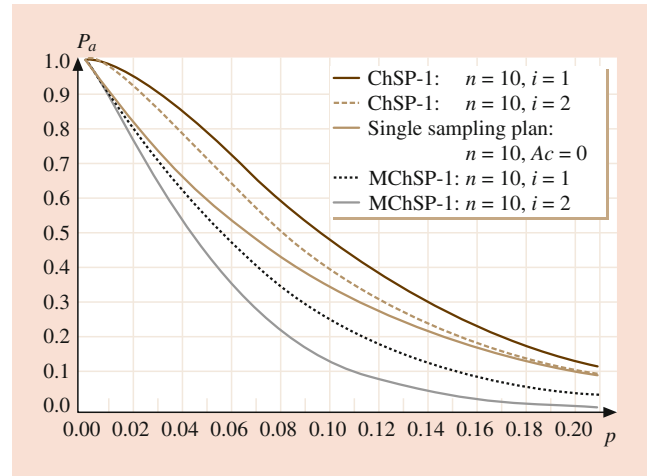


Fig. 12.4 Comparison of OC curves of ChSP-1 and MChSP-1 plans

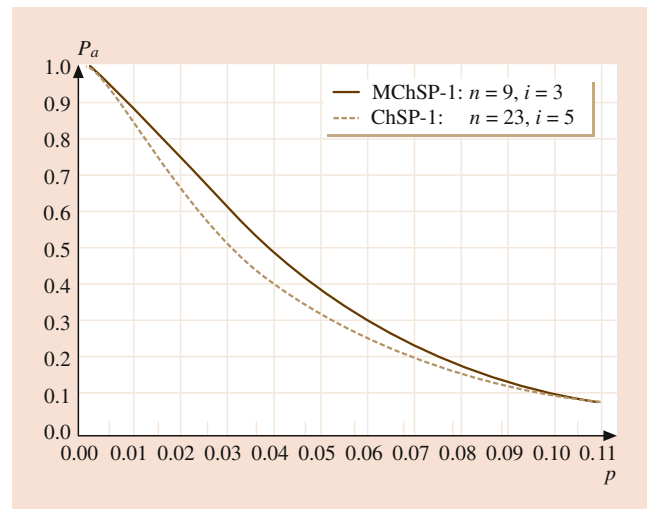


Fig. 12.5 OC curves of matched ChSP-1 and MChSP-1 plans

levels when compared to the OC curve of the zero acceptance number single sampling plan. The ChSP-1 plan, on the other hand, increases the probability of acceptance at good-quality levels but maintains the probability of acceptance at poor-quality levels. In order to compare the two sampling plans, we need to match them. That is, we need to design sampling plans whose OC curves approximately pass through two prescribed points such as $(AQL, 1 - \alpha)$ and (LQL, β) . Figure 12.5 gives such a comparison and establishes that MChSP-1 plan is efficient in requiring a very small sample size compared to the ChSP-1 plan. A two-stage chain sampling plan would generally require a sample size equal to or more than the sample size of a zero acceptance single sampling plan. The MChSP-1 plan will however require a sample size smaller than the zero acceptance number plan.

12.6 Chain Sampling and Deferred Sentencing

Like chain sampling plans, there are other plans that use the results of neighboring lots for taking a conditional decision of acceptance or rejection. Plans that make use of past lot results are either called chain or dependent sampling plans. Similarly plans that make use of future lot results are known as deferred sentencing plans. These plans have a strategy of accepting the lots conditionally based on the neighboring lot quality history and hence referred to as *conditional* sampling plans. We will briefly review several such conditional sampling plans available in the literature.

Contrary to chain sampling plans that make use of past lot results, deferred sentencing plans use future lot results. The idea of deferred sentencing was first published in a paper by Anscombe et al. [39]. The first simplest type deferred sentencing scheme of [39] requires the produced units be split into small size lots, and one item is selected from each lot for inspection. The lot sentencing rule is that whenever Y nonconforming units are found out of X or fewer consecutive lots tested, all such cluster of consecutive lots starting from the lot that resulted the first nonconforming unit to the lot that resulted the Y th nonconforming unit are rejected. Lots not rejected by this rule are accepted. This rule is further explained in the following sentences. A run of good lots of length X will be accepted at once. If a nonconforming unit occurs, then the lot sentencing or disposition will be deferred until either further $(X - 1)$ lots have been tested or $(Y - 1)$ further nonconforming items are found, whichever occurs the sooner. At the outset, if the $(X - 1)$ succeeding lots result in fewer than $(Y - 1)$ nonconforming units, the lot that resulted the first nonconforming unit and any succeeding lots clear of nonconforming units will be accepted. As soon as Y nonconforming units occur in no more than X lots, all lots not so far sentenced will be rejected. Thus the lot disposition will sometimes be made at once and sometimes with a delay not exceeding $(X - 1)$ lots. Some of the lots to be rejected according to the sentencing rule may already have been rejected through the operation of the rule on a previous cluster of Y nonconforming units partly overlapping the one being considered. The actual number of new lots rejected under the deferred sentencing rule can be any number from 1 to X . Anscombe et al. [39] also considered modifications of the above deferred sentencing rule, including inspection of a sample of size more than one from each lot. Anscombe et al. [39] originally presented their scheme as an alternative to [40] continuous sampling plan of type CSP-1 which is primarily intended for the partial screening inspection of produced units directly (when lot formation is difficult).

The deferred sentencing idea was formally tailored into an acceptance sampling plan by Hill et al. [41]. The operating procedure of [41] scheme is described below:

1. From each lot, select a sample of size n . The lots are accepted as long as no nonconforming units are found in the samples. If one or more nonconforming units are found, the disposition of the current lot will be deferred until $(X - 1)$ succeeding lots are inspected.
2. If the cumulative number of nonconforming units for X consecutive lots is Y or more, then a second sample of size n is taken from each of the lots (beginning with the first lot and ending with the last batch that showed a nonconforming unit in the sequence of X nonconforming units). If there are less than Y nonconforming units in the X , accept all lots from the first up to but not including the next batch that showed a nonconforming unit. The decision on this batch will be deferred until $(X - 1)$ succeeding lots are inspected.

Hill et al. [41] also evaluated the OC function of some selected schemes and found them very economical when compared to the traditional sampling plans, including the sequential attribute sampling plan.

Wortham and Mogg [42] developed a dependent stage sampling plan which is operated under steady state as follows:

1. For each lot, draw a sample of size n and observe the number of nonconforming units d .
2. If $d \leq r$, accept the lot; if $d > r + b$, reject the lot. If $r + 1 \leq d \leq r + b$, accept the lot if the $(r + b + 1 - d)$ th previous lot was accepted; otherwise, reject the current lot.

Govindaraju [43] has observed that the OC function of $DSSP(r; b)$ is the same as the OC function of the repetitive group sampling (RGS) plan of [44]. This means that the existing design procedures for the RGS plan can also be used for the design of $DSSP(r; b)$ plan. The deferred state sampling plan of [45] has a similar operating procedure except in Step 2 in which when the current lot is accepted if the forthcoming $(r + b + 1 - d)$ th lot is accepted. The steady-state OC function of the dependent (deferred) stage sampling plan is given by

$$P_a(p) = \frac{P_{a,r}(p)}{1 - P_{a,r+b}(p) + P_{a,r}(p)}$$

where $P_{a,r}(p)$ is the OC function of the single sampling plan with acceptance number r and sample size n . Similarly $P_{a,r+b}(p)$ is the OC function of the single sampling plan with acceptance number $r + b$ and sample size n . A procedure for

the determination of plan for given AQL, α , LQL, and β was also developed by Vaerst [22].

Wortham and Baker [46] extended the dependent (deferred) state sampling into a multiple dependent (deferred) state (MDS) plan $MDS(r, b, m)$. The operating procedure of the $MDS(r, b, m)$ plan is given below:

1. For each lot, draw a sample of size n and observe the number of nonconforming units d .
2. If $d \leq r$, accept the lot; if $d > r + b$, reject the lot. If $r + 1 \leq d \leq r + b$, accept the lot if the consecutive m preceding lots were all accepted (consecutive m succeeding lots must be accepted for the deferred $MDS(r, b, m)$ plan).

The steady-state OC function of the $MDS(r, b, m)$ plan is given by the recursive equation

$$P_a(p) = P_{a,r}(p) + [P_{a,r+b}(p) - P_{a,r}(p)] (P_a(p))^m$$

Vaerst [47], Soundararajan and Vijayaraghavan [48], Kuralmani and Govindaraju [49], and Govindaraju and Subramani [50] provided detailed tables and procedures for the design of $MDS(r, b, m)$ plans for various requirements. Afshari and Gildeh [51] provided a procedure to design the MDS plan in a fuzzy environment such when testing is imperfect.

Vaerst [22, 47] modified the $MDS(r, b, m)$ plan to make it on par with the ChSP-1 plan. The operating procedure of the modified $MDS(r, b, m)$ plan, called $MDS-1(r, b, m)$, is given below:

1. For each lot, draw a sample of size n and observe the number of nonconforming units d .
2. If $d \leq r$ accept the lot; if $d > r + b$, reject the lot. If $r + 1 \leq d \leq r + b$, accept the lot if r or less nonconforming units are found in each of the consecutive m preceding (succeeding) lots.

When $r = 0$, $b = 1$, and $m = i$, $MDS-1(r, b, m)$ plan becomes the ChSP-1 plan. The OC function of the plan is given by recursive equation

$$P_a(p) = P_{a,r}(p) + [P_{a,r+b}(p) - P_{a,r}(p)] (P_{a,r}(p))^m$$

Vaerst [47, 52], and [53] provided detailed tables and procedures for the design of plans for various requirements. The major but an obvious shortcoming of the chain sampling plans is that since they use sample information from past lots for disposing the current lot, there is a tendency to reject the current lot of given good quality when the process quality is improving or accept the current lot of given bad quality when the process quality is deteriorating. Similar criticisms (in reverse) can be leveled against the deferred sentencing

plans. As mentioned earlier, [33] recognizing this disadvantage of chain sampling defined the ARL performance measures ARL_1 and ARL_2 . Recall that ARL_2 is the average number of lots that will be accepted as a function of the true fraction nonconforming. ARL_1 is the average number of lots accepted after an upward shift in the true fraction nonconforming occurred from the existing level. Stephens and Dodge [54] evaluated the performance of the two-stage chain sampling plans comparing the ARLs with matching single and double sampling plans having approximately the same OC curve. It was noted that the slightly poorer ARL property due to chaining of lot results is well compensated by the gain in sampling economy. For deferred sentencing schemes, [41] investigated trends as well as sudden changes in quality. It was found that the deferred sentencing schemes will discriminate better between fairly constant quality at one level and fairly constant quality at another level than will a lot-by-lot plan scheme with the same sample size. However, when quality varies considerably from lot to lot, the deferred sentencing scheme found to operate less satisfactorily, and in certain circumstances the discrimination between good and bad batches may even be less than for the traditional unconditional plans with the same sample size. Further the deferred sentencing scheme may pose problems of flow, supply storage space, and uneven workload (which is not a problem with chain sampling).

Cox [55] provided a more theoretical treatment and considered one-step forward and two-step backward schemes. The lot sentencing rules and inspection are modeled as a stochastic process and applied Bayes's theorem for the sentencing rule. He did recognize the complexity of modeling a multistage procedure. When the submitted lot fraction nonconforming vary, say when a trend exists, both chain and deferred sentencing rules have disadvantages. But this disadvantage can be overcome by combining chain and deferred sentencing rules into a single scheme. This idea was first suggested by Baker [56] in his dependent deferred state (DDS) plan. Osanaiye [57] provided a complete methodology of combining chain and deferred sentencing rules and developed the chain-deferred (ChDP) plan. The ChDP plan has two stages for lot disposition and its operating procedure is given below:

1. From lot numbered k , inspect n units and count the number of nonconforming units d_k . If $d_k \leq c_1$, accept lot numbered k . If $d_k > c_2$, reject lot numbered k . If $c_1 < d_k \leq c_2$, then combine the number of nonconforming units from the immediate succeeding and preceding samples, namely, d_{k-1} and d_{k+1} .
2. If $d_k \leq c$, accept the k th lot provided $d_k + d_{k-1} \leq c_3$ (chain approach). If $d_k > c$, accept the lot provided $d_k + d_{k+1} \leq c_3$ (deferred sentencing).

One possible choice of c is the average of c_1 and $c_3 + 1$. Osanaiye [57] also provided a comparison of ChDP with the traditional unconditional double sampling plans as the OC curves of the two types of plans are the same (but the ChDP plan utilizes the neighboring lot results). Shankar and Srivastava [58] and Shankar and Joseph [59] provided a GERT analysis of ChDP plans taking the approach of [10]. Shankar and Srivastava [60] discussed the selection ChDP plans using tables. Osanaiye [61] provided a multiple sampling plan extension of the ChDP plan (called MChDP plan). MChDP plan uses several neighboring lot results to achieve sampling economy.

Osanaiye [62] provided a practically useful discussion on the choice of conditional sampling plans considering autoregressive processes, inert processes (constant process quality shift), and linear trends in quality. Based on a simulation study, it was recommended that the chain-deferred schemes are the cheapest if either the cost of 100% inspection or sampling inspection is high. He recommended the use of the traditional single or double sampling plans only if the opportunity cost of rejected items is very high. Osanaiye and Alebiosu [63] considered the effect of inspection errors on dependent and deferred double sampling plans vis-à-vis ChDP plans. They observed that the chain-deferred plan in general has a greater tendency to reject nonconforming items than any other plans irrespective of the magnitude of any inspection error.

Many of the conditional sampling plans, which follow either the approach of chaining or deferring or both, have the same OC curve of a double (or a multiple) sampling plan. Exploiting this equivalence, [64] provided a general selection procedure for conditional sampling plans for given AQL and LQL. The plans considered include the conditional double sampling plan of ChSP-4A plans of [13], conditional double sampling plan of [14], link sampling plan of [15], and ChDP plan of [57]. A perusal of the operating ratio LQL/AQL of [64] tables reveal that these conditional sampling plans apply in all Type B situations because a wide range of discrimination between good and bad qualities is achievable. However, the sample sizes, even though smaller than the traditional unconditional plans, will not be as small as the zero acceptance number single sampling plans. This limits the application of the conditional sampling plans to this special purpose situation, where ChSP-1 or MChSP-1 plan suits the most.

Govindaraju [65] developed a conditional single sampling (CSS) plan, which has desirable properties for general applications as well as for costly or destructive testing. The operating procedure of the CSS plan is as follows:

1. From lot numbered k , select a sample of size n and observe the number of nonconforming units d_k .

2. Cumulate the number of nonconforming units observed for the current lot and the related lots. The related lots will be either past lots, future lots, or a combination depending on whether one is using dependent sampling or deferred sentencing. The lot under consideration is accepted if the total number of nonconforming units in the current lot and the m related lots is less than or equal to the acceptance number, Ac . If d_k is the number of nonconforming units recorded for the k th lot, the rule for disposition for the k th lot is as follows:

- a. For dependent or chain sampling, accept the lot if $(d_{k-m} + \dots + d_{k-1} + d_k) \leq Ac$; otherwise, reject the lot.
- b. For deferred sampling, accept the lot if $(d_k + d_{k+1} + \dots + d_{k+m}) \leq Ac$; otherwise, reject the lot.
- c. For dependent-deferred sampling, where m is desired to be even, accept the lot if $(d_{k-\frac{m}{2}} + \dots + d_k + \dots + d_{k+\frac{m}{2}}) \leq Ac$; otherwise, reject the lot.

Thus, the CSS plan has three parameters, the sample size n , acceptance number Ac , and number of related lot results used, m . As in the case of any dependent sampling procedure, dependent single sampling takes full effect only from the $(m + 1)^{\text{st}}$ lot. In order to maintain *equivalent* OC protection for the first m lots, additional sample mn units can be taken from each lot and the lot can be accepted if the total number of nonconforming units is less than or equal to Ac or additional samples of size $(m + 1 - i)n$ can be taken for the i th lot ($i = 1, 2, \dots, m$) and the same decision rule be applied. In either case, the results of the additional samples should not be used for lot disposition from lot $(m + 1)$. Govindaraju [65] has shown that the CSS plans require much smaller sample sizes than all other conditional sampling plans. In case of trends in quality, the CSS plan can also be operated as a chain-deferred plan, and this will ensure that the changes in lot qualities are somewhat averaged out.

12.7 Comparison of Chain Sampling with Switching Systems

Dodge [66] first proposed the quick switching sampling (QSS) system which basically consists of two intensities of inspection, i.e., normal (N) and tightened (T) plans. Romboski [67] investigated the QSS system and introduced several modifications of the original QSS system. Under the QSS system, if a lot is rejected under the normal inspection, a switch tightened inspection will be made; otherwise, normal inspection shall continue. If a lot is accepted under the tightened inspection, then the normal inspection will be restored; otherwise, tightened inspection will be continued. For a review of quick switching systems, see [68] or [69].

Taylor [68] introduced a new switch number to the original QSS-1 system of [67] and compared it with the chain sampling plans. When the sample sizes of normal and tightened plans are equal, the quick switching systems and the two-stage chain sampling plans were found to give nearly identical performance. Taylor's comparison is only valid for a general situation where acceptance numbers of greater than zero are used. For costly or destructive testing, acceptance numbers are kept at zero for achieving minimum sample sizes. In such situations, both ChSP-1 and ChSP-(0,1) plans will fare poorly against other comparable schemes when the incoming quality is at AQL. This fact is explained in the following paragraph using an example.

For costly or destructive testing, a quick switching system employing zero acceptance number was studied in [70] and [71]. Under this scheme, the normal inspection plan has a sample size of n_N units, while the tightened inspection plan has a higher sample size $n_T (> n_N)$. The acceptance number is kept at zero for both normal and tightened inspections. The switching rule is that a rejection under the normal plan ($n_N, 0$) will invoke the tightened plan ($n_T, 0$). An acceptance under the ($n_T, 0$) plan will revert back to normal inspection. This QSS system, designated as of type QSS-1($n_N, n_T; 0$), can be used in place of the ChSP-1 and ChSP-(0,1) plans. Let AQL = 1%, $\alpha = 5%$, LQL = 15%, and $\beta = 10%$. The ChSP-1 plan for the prescribed AQL and LQL conditions is found as $n = 15$ and $i = 2$ (from Table 12.1 of this chapter). The matching QSS-1 system for the prescribed AQL and LQL conditions can be found as QSS-1($n_N = 5, n_T = 19$) from the tables given in [70] or [72]. At good-quality levels, the normal inspection plan will require sampling only 5 units. Only at poor-quality levels, 19 units will be sampled under the QSS system. Hence, it is obvious that Dodge's chain sampling approach is not truly economical at good-quality levels but fares well at poor-quality levels. However, if the modified chain sampling plan MChSP-1 plan of [38] is used, then the sample size needed will be only 3 units (and i , the number of related lot results to be used is fixed at 7 or 8). The other alternative is to use the chained quick switching system proposed in [73]. For a detailed discussion of this approach, consult [74].

A more general two-plan system having zero acceptance numbers for the tightened and normal plans was studied by Calvin [75], Soundararajan and Vijayaraghavan [76], and Subramani and Govindaraju [77]. Calvin's TNT scheme uses zero acceptance numbers for normal and tightened inspection and employs the switching rules of [78], which is also roughly employed in [79]. The operating procedure of the TNT scheme, designated as TNT($n_N, n_T; 0$), is given below:

1. Start with the tightened inspection plan ($n_T, 0$). Switch to normal inspection (Step 2) when t lots in a row are

accepted; otherwise, continue with the tightened inspection plan.

2. Apply the normal inspection plan ($n_N, 0$). Switch to the tightened plan if a lot rejection is followed by another lot rejection within the next s lots.

Using the tables of [80], the zero acceptance number TNT($n_N, n_T; 0$) plan for given AQL = 1%, $\alpha = 5%$, LQL = 15%, and $\beta = 10%$ is found as TNT($n_N = 5, n_T = 16; A_c = 0$). We again find that the MChSP-1 plan calls for a smaller sample size when compared to Calvin's zero acceptance number TNT plan.

Skip-lot sampling plans of [81] and [82] are based on skipping of sampling inspection of lots on the evidence of good-quality history. For a detailed discussion of skip-lot sampling, [32] may be consulted. In skip-lot sampling plan of type SkSP-2 of [82], once m successive lots are accepted under the reference plan, the chosen reference sampling plan is applied only for a fraction of the time. Govindaraju [83] studied the employment of the zero acceptance number plan as a reference plan (among several other reference sampling plans) in the skip-lot context. For given AQL = 1%, $\alpha = 5%$, LQL = 15%, and $\beta = 10%$, the SkSP-2 plan with a zero acceptance number reference plan is found as $n = 15$, $m = 6$, and $f \approx 1/5$. Hence, the matching ChSP-1 plan $n = 15$ and $i = 2$ is not economical at good-quality levels when compared to the SkSP-2 plan $n = 15$, $m = 6$, and $f \approx 1/5$. This is because the SkSP-2 plan requires the zero acceptance number reference plan with a sample of size 15 to be applied only to one in every five lots submitted for inspection once six consecutive lots are accepted under the reference single sampling plan ($n = 10, A_c = 0$). However, the modified MChSP-1 plan is more economical at poor-quality levels when compared to the SkSP-2 plan. Both plans require about the same sampling effort at good-quality levels.

12.8 Chain Sampling for Variable Inspection

The main assumption made for the various types of chain sampling plans and other attribute schemes such as deferred-dependent plans and quick switching systems is that the fraction nonconforming p in a series of lots roughly remains a constant. No other distributional assumptions are made for attribute sampling inspection plans. If the assumption of constant p for a series of lots is violated, there can be a delay in detection of a change in p . This delay is measured using the difference ($ARL_1 - ARL_2$) (see Sect. 12.4). However, if the rule of chaining lot inspection results is modified as a chain-deferred rule, the overall producer's and consumer's risk will remain the same even if there is a linear trend in p in a series of lots (see the discussion in Sect. 12.5). If the distribution of the

Table 12.3 Limits for deciding unsatisfactory variable plans

n_σ	$k_{\sigma l}$	n_σ	$k_{\sigma l}$	n_σ	$k_{\sigma l}$	n_σ	$k_{\sigma l}$
1	0	16	2.3642	31	3.3970	46	4.1830
2	0.4458	17	2.4465	32	3.4549	47	4.2302
3	0.7280	18	2.5262	33	3.5119	48	4.2769
4	0.9457	19	2.6034	34	3.5680	49	4.3231
5	1.1278	20	2.6785	35	3.6232	50	4.3688
6	1.2869	21	2.7515	36	3.6776	51	4.4140
7	1.4297	22	2.8227	37	3.7312	52	4.4588
8	1.5603	23	2.8921	38	3.7841	53	4.5032
9	1.6812	24	2.9599	39	3.8362	54	4.5471
10	1.7943	25	3.0262	40	3.8876	55	4.5905
11	1.9009	26	3.0910	41	3.9384	56	4.6336
12	2.0020	27	3.1546	42	3.9885	57	4.6763
13	2.0983	28	3.2169	43	4.0380	58	4.7186
14	2.1904	29	3.2780	44	4.0869	59	4.7605
15	2.2789	30	3.3380	45	4.1352	60	4.8021

quality characteristic of interest is known, variable inspection becomes feasible. Govindaraju and Balamurali [84] extended the idea of chain sampling to variable inspection assuming normality. This approach is particularly useful when testing is costly or destructive provided the quality variable is measurable on a continuous scale and known to be normally distributed. It is well known that the variable plans do call for a very low sample sizes when compared to the attribute plans. However, not all variable plans possess a satisfactory OC curve as shown by Govindaraju and Kuralmani [85]. Often, a variable plan is unsatisfactory if the acceptability constant is too large particularly when the sample size is small. Only in such cases, it is necessary to follow the chain sampling approach to improve upon the OC curve of the variable plan. Table 12.3 is useful for deciding whether a given variables sampling plan has a satisfactory OC curve or not. If the acceptability constant k_σ of a known sigma variables plan exceeds $k_{\sigma l}$, then the plan is deemed to have an unsatisfactory OC curve like an $A_c = 0$ attribute plan.

The operating procedure of the chain sampling plan for variables inspection is as follows:

1. Draw a random sample of size n_σ , say $(x_1, x_2, \dots, x_{n_\sigma})$, and then compute $v = (U - \bar{X})/\sigma$ where $\bar{X} = \sum_{i=1}^{n_\sigma} x_i/n_\sigma$.
2. Accept the lot if $v \geq k_\sigma$ and reject if $v < k'_\sigma$. If $k'_\sigma \leq v < k_\sigma$, accept the lot provided the preceding i lots were accepted on the condition that $v \geq k_\sigma$.

Thus the variable chain sampling plan has four parameters, namely, the sample size n_σ ; the acceptability constants k_σ and k'_σ ($< k_\sigma$); and i , the number of preceding lots used for conditionally accepting the lot. The OC function of this plan is given by $P_a(p) = P_V + (P'_V - P_V)P_V^i$ where

$P_V = \Pr(v \geq k_\sigma)$ is the probability of accepting the lot under the variables plan (n_σ, k_σ) and $P'_V = \Pr(v \geq k'_\sigma)$ is the probability of accepting the lot under the variables plan (n_σ, k'_σ) . Even though the above operating procedure of the variables chain sampling plan is of a general nature, it would be appropriate to fix $k'_\sigma = k_{\sigma l}$. For example, suppose that a variable plan with $n_\sigma = 5$ and $k_\sigma = 2.46$ is currently under use. From Table 12.3, the limit for the undesirable acceptability constant $k_{\sigma l}$ for $n_\sigma = 5$ is obtained as 1.1278. Since the regular acceptability constant $k_\sigma = 2.26$ is greater than $k'_\sigma = 1.1278$, the variable plan can be declared to possess an unsatisfactory OC curve. Hence, it is desirable to chain the results of neighboring lots to improve upon the shape of the OC curve of the variable plan $(n_\sigma = 5, k_\sigma = 2.26)$, that is, the variable plan currently under use will be operated as a chain sampling plan fixing i as four. A more detailed procedure on designing chain sampling for variable inspection, including the case when sigma is unknown, is available in [84]. The chain sampling for variables will be particularly useful when inspection costs are prohibitively high, and the quality characteristic is measurable on a continuous scale.

Luca [86] extended the modified chain sampling plan MChSP-1 of [38] for variable inspection. The main advantage of the MChSP-1 is the reduction of sample size compared to the ChSP-1 plan. While the variable chain sampling approach reduces the amount inspection substantially, the variable inspection based on the modified chain sampling plan rule can achieve some marginal gains.

The idea of using chain sampling rules for variables inspection proved popular in the recent years, and a large number of plans using conditional lot disposition rules appeared in the literature. Balamurali and Jun [87] extended the variable chain sampling idea to MDS sampling plans based on the normal distribution. Balamurali et al. [88] employed the Weibull distribution so that the plans can be used for lifetime assurance (see [89] and [90]). Aslam et al. [91] extended the MDS rule to a lifetime characteristic following the Burr XII distribution.

Arizono et al. [92] first coined the idea of employing the process capability index for the design of a variables inspection plan. Wu et al. [93] extended the work of [87] to MDS sampling plan using the capability index C_{pk} for normally distributed processes with two-sided specification limits (see [94] who also developed a variables MDS sampling plan for lot sentencing based on the process capability index) (also see [95]). Wu et al. [96] employed the quick switching rules and presented variable quick switching sampling (VQSS) system based on the C_{pk} index. There are numerous papers employing rules based on chaining lot results which employ various process capability indices, and the reader is suggested to refer to the review given in [96]. Kurniati et al. [97] employed the TNT plan rules with a unilateral specification

limit based on one-sided capability indices. Yan et al. [98] extended the MDS rules to assure protection of coefficient of variation of a normally distributed quality characteristic.

12.9 Chain Sampling and CUSUM

In this section, we will discuss some of the interesting relationships between the cumulative sum (CUSUM) approach of [99, 100] and chain sampling approach of [1]. As explained in [101], the CUSUM approach for Gaussian processes is largely popular in the area of statistical process control (SPC), but [99] proposed it to be used with attribute (binomial) inspection problems arising in acceptance sampling as well. Page [99] compares the CUSUM-based inspection scheme with the deferred sentencing schemes of [39] and the continuous sampling plan CSP-1 of [40] for evaluating their relative performance. In fact Dodge's CSP-1 plan forms the theoretical basis for the ChSP-1 chain sampling plan. A more formal acceptance sampling scheme based on the one-sided CUSUM for lot-by-lot inspection was proposed in [102]. Beattie's plan calls for drawing a random sample of size n from each lot and observing the number of nonconforming units d . For each lot, a CUSUM value is calculated for a given slack parameter k . If the computed CUSUM is within the decision interval $(0, h)$, then the lot is accepted. If the CUSUM is within the *return interval* (h, h') , then the lot is rejected. If the CUSUM falls below zero, it is reset to zero. Similarly, if the CUSUM exceeds $h + h'$, it is reset to $h + h'$. In other words, for the j th lot, the plotted CUSUM can be succinctly defined as $S_j = \min(h + h', \max(d_j - k + S_{j-1}))$ with $S_0 = 0$. Beattie's plan is easily implemented using the typical number of nonconforming units CUSUM chart for lot-by-lot inspection (see Fig. 12.6).

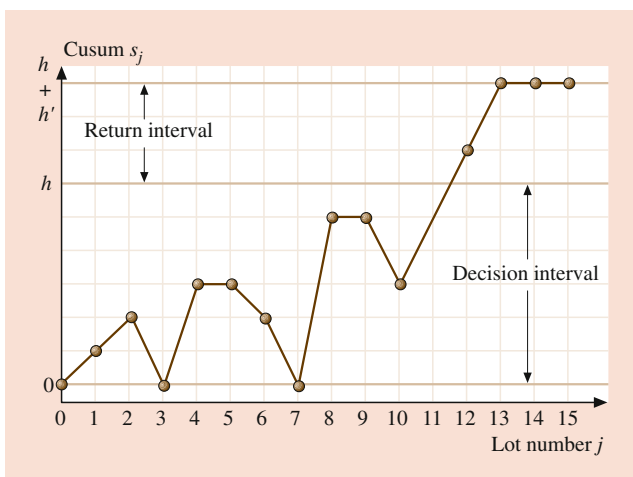


Fig. 12.6 Beattie's CUSUM acceptance sampling plan

Prairie and Zimmer [103] provided detailed tables and nomographs for the selection of Beattie's CUSUM acceptance sampling plan. An application is also reported in [104].

Beattie [105] introduced a "two-stage" semicontinuous plan where the CUSUM approach is followed and the product is accepted as long as the CUSUM S_j is within the decision interval $(0, h)$. For product falling in the return interval (h, h') , an acceptance sampling plan such as the single or double sampling plan is used for lot disposition. Beattie [105] compared the "two-stage" semicontinuous plan with the ChSP-4A plan of [13] and the deferred sentencing scheme of [41]. It is also remarked in [105] that chain sampling plans (ChSP-4A type) call for steady rate of sampling and simple to administer. The two-stage semicontinuous sampling plan achieved some gain in the average sample number at good-quality levels but is more difficult to administer. The two-stage semicontinuous plan also requires more sample size than ChSP-4A plans when the true quality is poorer than acceptable levels.

We will now explore an interesting equivalence between the ChSP-1 plan and a CUSUM scheme intended for high-yield or low-fraction nonconforming production processes for which the traditional p or np control charts are not useful. Lucas [106] gave a signal rule for lack of statistical control if there are two or more counts within an interval of t samples. In case of low-process fraction nonconforming, this means that if two or more nonconforming units are observed in any t consecutive samples or less, a signal for an upward shift in the process fraction level is obtained. It should be noted that if two or more nonconforming units are found even in the same sample, a signal for lack of statistical control will be obtained. Govindaraju and Lai [107] discussed the design given in [106] and provided a method of obtaining the parameters n (the subgroup or sample size) and t (the maximum number of consecutive samples considered for a signal). Lucas [106] has shown that his signal rule is equivalent to a CUSUM scheme having a reference value $k = 1/t$ and decision interval $(0, h = 1)$ for detecting an increase in the process count level. It was also shown that a fast initial response (FIR) feature can be added to the CUSUM scheme (see [108]) with an additional subrule of signaling lack of statistical control if the first count occurs before the t th sample. This FIR CUSUM scheme has a head start of $S_0 = 1 - k$ with $k = 1/t$ and $h = 1$. Consider the ChSP-1 plan of [1] which rejects a lot if two or more counts (of nonconformity or nonconforming units) occur but allows acceptance of the lot if no counts occur or a single count is preceded by t (the symbol i was used before) lots having samples with no counts. If the decision of rejecting a lot is translated as the decision of declaring the process to be not in statistical control, then it is seen that Lucas's scheme and the ChSP-1 plan are the same. This equivalence will be further clearer if one considers the operation of two-stage chain sampling plan ChSP-(0,1) of [27] given in Sect. 12.4. When

$k_2 = k_1 + 1$, the ChSP-(0,1) plan is equivalent to the ChSP-1 plan with $t = k_1$. Hence, it can also be noted that the subrule of not allowing any count for the first t samples suggested for the FIR CUSUM scheme of [106] is an inherent feature of the two-stage chain sampling scheme. This means that the ChSP-1 plan is equivalent to the FIR CUSUM scheme with the head start $(1 - k)$ with $k = 1/t$ and $h = 1$.

12.10 Other Interesting Extensions

If homogeneous lot formation is difficult, and random sampling from the lot is harder, the binomial assumption may not be valid. Gao and Tang [109] considered the chain sampling rule when testing for conformance status which exhibits correlation between successive units. A two-state Markov chain model for the correlation within each sample was assumed as against a two-state Markov chain model for the fraction nonconforming p . In acceptance sampling, random samples are taken; hence, the correlation within the sample is not an issue. The proposal given in [109] may only apply when random samples are difficult to obtain.

Mixed sampling plans are two-phase sampling plans in which both variable quality characteristics and attribute quality measures are used in deciding the acceptance or rejection of the lot. Baker and Thomas [110] reported the application of chain sampling for acceptance testing for armor packages. Their procedure uses chain sampling for testing structural integrity (attributes inspection), and a variable sampling plan is used for testing penetration depth quality characteristic. Baker and Thomas [110] also suggested the simultaneous use of control charts along with their proposed acceptance sampling procedures. Suresh and Devaarul [111] proposed a more formal mixed acceptance sampling plan where a chain sampling plan is used for the attribute phase. Suresh and Devaarul [111] also obtained the OC function for their mixed plan and discussed various selection procedures. For controlling multidimensional characteristics, [112] developed multidimensional mixed sampling plans (MDMSP). This plan handles several quality characteristics during the variable phase of the plan, while the attribute sampling phase can be based on chain sampling or other attribute plans. Aslam et al. [113] presented mixed multiple dependent state sampling plans based on the popular process capability index. Balamurali [114] and Usha and Balamurali [115] extended the mixed chain sampling approach using the process capability index C_{pk} . In a similar vein, another form of mixed modified chain sampling plan is given in [116]. Balamurali et al. [117] generalized the mixed sampling incorporating the multiple dependent state sentencing rule. Balamurali and Usha [118] also extended the mixed sampling incorporating the quick switching system rules. These approaches would

further reduce the sampling effort but no investigation was done on the delay in detection of a shift.

In some situations, it is desirable to adopt three attribute classes where items are classified into three categories, namely, good, marginal, and bad (see [119]). Shankar et al. [120] developed three class chain sampling plans and derived various performance measures through GERT approach and also discussed their design.

Suresh and Deepa [121] provided a discussion on formulating a chain sampling plan given a gamma or beta prior distribution for product quality. Tables are for the selection of the plans and examples are also provided by Suresh and Deepa [121]. Latha and Jeyabharathi [122] considered the beta-binomial process for the operation of the ChSP-1 plan. The main limitation of this approach is that the assumption of nearly constant process fraction nonconforming needed for chain sampling is not fulfilled. As a result, the $(ARL_1 - ARL_2)$ delay will be higher. A number of other lot chaining rules were also proposed such as the rule given in [123] where the number of preceding lot results employed depends on the number of nonconforming results found in the current sample.

Tang and Cheong [124] extended the idea of the chain inspection procedure to enhance its sensitivity in detecting a process shift while monitoring high-quality processes with low fraction nonconforming. The MDS plan rules are also used in control charting (see [125] and [126]).

12.11 Concluding Remarks

Chain sampling attribute inspection plans for a series of lots can be implemented without any distributional assumption. The main requirement is that the fraction nonconforming p in the series of lots submitted for inspection is fairly constant. If there is any linear trend in p , the chain sampling inspection can be modified as a chain-deferred inspection procedure so that the producer's and consumer's risks are maintained the same for the series of lots inspected. If the chain sampling rule is applied for variable inspection, distributional assumptions such as a normality must be met. For costly and destructive testing, the sample size must be kept small. Small sample sizes such as five can be effective when lot results are chained under variable inspection. If the normality assumption is suspect, it is desirable to employ the regular attribute chain sampling in order to assure the set risks.

This chapter largely reviewed the methodology of chain sampling inspection of quality in a series of lots. Various extensions of the original chain sampling plan ChSP-1 of [1] and modifications are briefly reviewed. Chain sampling approach is primarily useful for costly or destructive testing, where small sample sizes are preferred. Chain sampling achieves greater sampling economy when it is combined with

the approach of deferred sentencing so that the combined plan can be used for any general situations. This chapter does not cover designing of chain sampling plans in any great detail. One may consult textbooks such as [74] or [32, 127] for detailed tables. A large number of papers primarily dealing with the design of chain sampling plans are available only in journals, and some of them are listed as references. It is often remarked that designing sampling plans is more of an art than a science. There are statistical, engineering, and other administrative aspects that are to be taken into account for successful implementation of any sampling inspection plan, including the chain sampling plans. For example, the sample size may be fixed due to administrative and other reasons. Given this limitation, what sampling plan should be used requires careful consideration. Several candidate sampling plans, including chain sampling plans, must be sought first, and then the selection of a particular type of plan must be made based on the performance measures such as the OC curve, etc. Sampling plans that make use of related lot results must also be investigated for their performance against trend in the submitted lot quality as well as sudden jumps. The effectiveness of a chosen plan or a sampling scheme must be monitored over time for a series of batches. The severity of inspection is reduced with the chaining-related lot results, but consumer protection should receive more attention by way of minimizing the delay in detection of a poor-quality lot submission.

References

- Dodge, H.F.: Chain sampling inspection plan. *Indust. Qual. Contr.* **11**, 10–13 (1955)
- Cameron, J.M.: Tables for constructing and for computing the operating characteristics of single sampling plans. *Indust. Qual. Contr.* **9**(1), 37–39 (1952)
- Guenther, W.C.: Use of the binomial, hypergeometric and Poisson tables to obtain sampling plans. *J. Qual. Technol.* **1**(2), 105–109 (1969)
- Hahn, G.J.: Minimum size sampling plans. *J. Qual. Technol.* **6**(3), 121–127 (1974)
- Clark, C.: OC curves for ChSP-1. Chain sampling plans. *Indust. Qual. Contr.* **17**(4), 10–12 (1960)
- Liebman, B., Hawley, F.: Small acceptance number plans for use in Military Standard 105D. *J. Qual. Technol.* **16**(4), 219–231 (1984)
- Soundararajan, V.: Procedures and tables for construction and selection of chain sampling plans (ChSP-1) Part 1. *J. Qual. Technol.* **10**(2), 56–60 (1978)
- Soundararajan, V.: Procedures and tables for construction and selection of chain sampling plans (ChSP-1) Part 2: Tables for selection of chain sampling plans. *J. Qual. Technol.* **10**(3), 99–103 (1978)
- Govindaraju, K.: Selection of ChSP-1 chain sampling plans for given acceptable quality level and limiting quality level. *Commun. Stat.-Theor. M.* **19**(6), 2179–2190 (1990)
- Ohta, H.: GERT analysis of chain sampling inspection plans. *Bull. Uni. Osaka Prefecture, Sec. A, Eng. Nat. Sci.* **27**, 67–174 (1979)
- Raju, C., Jothikumar, J.: A design of chain sampling plan ChSP-1 based on Kullback-Leibler information. *J. Appl. Statist.* **21**(3), 153–160 (1994)
- Govindaraju, K.: Selection of minimum ATI ChSP-1 chain sampling plans. *IAPQR Trans.* **14**, 91–97 (1990)
- Frishman, F.: An extended chain sampling inspection plan. *Indust. Qual. Contr.* **17**, 10–12 (1960)
- Baker, R.C., Brobst, R.W.: Conditional double sampling. *J. Qual. Technol.* **10**(4), 150–154 (1978)
- Harishchandra, K., Sriveakataramana, P.: Link sampling for attributes. *Commun. Stat.-Theor. M.* **11**(16), 1855–1868 (1982)
- Raju, C.: On equivalence of OC functions of certain conditional sampling plans. *Commun. Stat.- Simul. C.* **21**(4), 961–969 (1992)
- Raju, C.: Procedures and tables for the construction and selection of chain sampling plans ChSP-4A(c_1, c_2) r —Part 1. *J. Appl. Statist.* **18**(3), 361–381 (1991)
- Raju, C.: Procedures and tables for the construction and selection of chain sampling plans ChSP-4A(c_1, c_2) r —Part 2. *J. Appl. Statist.* **19**(1), 125–140 (1992)
- Raju, C., Murthy, M.N.N.: Procedures and tables for the construction and selection of chain sampling plans ChSP-4A(c_1, c_2) r —Part 3. *J. Appl. Statist.* **20**(4), 495–511 (1993)
- Raju, C., Murthy, M.N.N.: Procedures and tables for the construction and selection of chain sampling plans ChSP-4A(c_1, c_2) r —Part 4. *J. Appl. Statist.* **22**(2), 261–271 (1995)
- Raju, C., Jothikumar, J.: Procedures and tables for the construction and selection of chain sampling plans ChSP-4A(c_1, c_2) r —Part 5. *J. Appl. Statist.* **24**(1), 49–76 (1997)
- Vaerst, R.: About the determination of minimum conditional attribute acceptance sampling procedures Ph.D. Thesis. University of Siegen, Berlin (1981)
- Vidya, R.: Minimum variance and VOQL chain sampling plans—ChSP-4(c_1, c_2). *Commun. Stat.- Simul. C.* **37**(7), 1466–1478 (2008)
- Bagchi, S.B.: An extension of chain sampling plan. *IAPQR Trans* **1**, 19–22 (1976)
- Subramani, K., Govindaraju, K.: Bagchi's extended two stage chain sampling plan. *IAPQR Trans* **19**, 79–83 (1994)
- Gao, Y.: Studies on chain sampling schemes in quality and reliability engineering Ph.D. Thesis. National University of Singapore, Singapore (2003)
- Dodge, H.F., Stephens, K.S.: Some new chain sampling inspection plans. *Indust. Qual. Contr.* **23**, 61–67 (1966)
- Stephens, K.S., Dodge, H.F.: Two-Stage chain sampling inspection plans with different sample sizes in the two stages. *J. Qual. Technol.* **8**(4), 209–224 (1976)
- Soundararajan, V., Govindaraju, K.: Construction and selection of chain sampling plans ChSP-(0,1). *J. Qual. Technol.* **15**(4), 180–185 (1983)
- Subramani, K.G.K.: Selection of ChSP-(0,1) plans for given IQL and MAPD. *Int. J. Qual. Rel. Manag.* **8**, 39–45 (1991)
- Govindaraju, K., Subramani, K.: Selection of chain sampling plans ChSP-1 and ChSP-(0,1) for given acceptable quality level and limiting quality level. *Am. J. Math. Manag. Sci.* **13**(1–2), 123–136 (1993)
- Stephens, K.S.: How to Perform Skip-lot and Chain sampling, vol.4. In: *ASQ Basic References in Quality Control*. ASQ Press, New York (1995)
- Stephens, K.S., Dodge, H.F.: Evaluation of response characteristics of chain sampling inspection plans Technical Report No. N-25, p. 53. The Statistics Center, Rutgers- The State University, Wisconsin (1967)
- Stephens, K.S., Dodge, H.F.: An application of Markov chains for the evaluation of the operating characteristics of chain sampling inspection plans. *The QR Journal* **1**, 131–138 (1974)

35. Raju, C., Murthy, M.N.: Two-stage chain sampling plans ChSP-(0,2) and ChSP-(1,2) - Part 1. *Commun. Stat.- Simul. C.* **25**(2), 557–572 (1996)
36. Jothikumar, J., Raju, C.: Two stage chain sampling plans ChSP-(0,2) and ChSP-(1,2), - Part 2. *Commun. Stat.- Simul. C.* **25**(3), 817–834 (1996)
37. Raju, C.: Three-stage chain sampling plans. *Commun. Stat.-Theor. M.* **20**(5–6), 1777–1801 (1991)
38. Govindaraju, K., Lai, C.D.: A modified ChSP-1 chain sampling plan, MChSP-1, with very small sample sizes. *Am. J. Math. Manag. Sci.* **18**(3–4), 343–358 (1998)
39. Anscombe, F.J., Godwin, H.J., Plackett, R.L.: Methods of deferred sentencing in testing the fraction defective of a continuous output. *J. R. Statist. Soc. Suppl.* **9**(2), 198–217 (1947)
40. Dodge, H.F.: A sampling inspection plan for continuous production. *Ann. Math. Stat.* **14**(3), 264–279 (1943)
41. Hill, I.D., Horsnell, G., Warner, B.T.: Deferred sentencing schemes. *J. R. Statist. Soc. Ser. C (Appl. Statist.)* **8**(2), 76–91 (1959)
42. Wortham, A.W., Mogg, J.M.: Dependent stage sampling inspection. *Int. J. Prod. Res.* **8**(4), 385–395 (1970)
43. Govindaraju, K.: An interesting observation in acceptance sampling. *Econ. Qual. Contr.* **2**, 89–92 (1987)
44. Sherman, R.E.: Design and evaluation of a repetitive group sampling plan. *Technometrics* **7**(1), 11–21 (1965)
45. Wortham, A.W., Baker, R.C.: Deferred state sampling procedures. In: *Annual Assurance Science*, pp. 65–70. Annual Symposium on Reliability, Washington (1971)
46. Wortham, A.W., Baker, R.C.: Multiple deferred state sampling inspection. *Int. J. Prod. Res.* **14**(6), 719–731 (1976)
47. Vaerst, R.: A method to determine MDS sampling plans (in German). *Meth. Oper. Res.* **37**, 477–485 (1980)
48. Soundararajan, V., Vijayaraghavan, R.: Construction and selection of multiple dependent (deferred) state sampling plan. *J. Appl. Statist.* **17**(3), 397–409 (1990)
49. Kuralmani, V., Govindaraju, K.: Selection of multiple deferred (dependent) state sampling plans. *Commun. Stat.-Theor. M.* **21**(5), 1339–1366 (1992)
50. Govindaraju, K., Subramani, K.: Selection of multiple deferred (dependent) state sampling plans for given acceptable quality level and limiting quality level. *J. Appl. Statist.* **20**(3), 423–428 (1993)
51. Afshari, R., Gildeh, B.S.: Designing a multiple deferred state attribute sampling plan in a fuzzy environment. *Am. J. Math. Manag. Sci.* **36**(4), 328–345 (2017)
52. Soundararajan, V., Vijayaraghavan, R.: On designing multiple deferred state sampling (MDS-1 (0, 2)) plans involving minimum risks. *J. Appl. Statist.* **16**(1), 87–94 (1989)
53. Govindaraju, K., Subramani, K.: Selection of multiple deferred state MDS-1 sampling plans for given acceptable quality level and limiting quality level involving minimum risks. *J. Appl. Statist.* **17**(3), 427–434 (1990)
54. Stephens, K.S., Dodge, H.F.: Comparison of chain sampling plans with single and double sampling plans. *J. Qual. Technol.* **8**(1), 24–33 (1976)
55. Cox, D.R.: Serial sampling acceptance schemes derived from Bayes's theorem. *Technometrics* **2**(3), 353–360 (1960)
56. Baker, R.C.: Dependent-deferred state attribute acceptance sampling Ph.D. Thesis. Texas A&M University, Arlington (1971)
57. Osanaiye, P.A.: Chain-Deferred inspection plans. *J. R. Statist. Soc. Ser. C (Appl. Statist.)* **32**(1), 19–24 (1983)
58. Shankar, G., Srivastava, R.K.: GERT Analysis of two-stage deferred sampling plan. *Metron* **54**, 181–193 (1996)
59. Shankar, G., Joseph, S.: GERT analysis of Chain-deferred (ChDF-2) sampling plan. *IAPQR Trans.* **21**, 119–124 (1996)
60. Shankar, G., Srivastava, R.K.: Procedures and tables for construction and selection of chain-deferred (ChDF-2) sampling plan. *Int. J. Manag. Sys.* **12**, 151–156 (1996)
61. Osanaiye, P.A.: Multiple chain-deferred inspection plans and their compatibility with the multiple plans in MIL-STD-105D and equivalent schemes. *J. Appl. Statist.* **12**(1), 71–81 (1985)
62. Osanaiye, P.A.: An economic choice of sampling inspection plans under varying process quality. *J. R. Statist. Soc. Ser. C (Appl. Statist.)* **38**(2), 301–308 (1989)
63. Osanaiye, P.A., Alebiosu, S.A.: Effects of industrial inspection errors on some plans that utilise the surrounding lot information. *J. Appl. Statist.* **15**(3), 295–304 (1988)
64. Kuralmani, V., Govindaraju, K.: Selection of conditional sampling plans for given AQL and LQL. *J. Appl. Statist.* **20**(4), 467–479 (1993)
65. Govindaraju, K.: Conditional single sampling procedure. *Commun. Stat.-Theor. M.* **26**(5), 1215–1225 (1997)
66. Dodge, H.F.: A new dual system of acceptance sampling Technical Report No. N-16. The Statistics Center, Rutgers- The State University, Wisconsin, NJ (1967)
67. Romboski, L.D.: An investigation of quick switching acceptance sampling systems Ph.D. Thesis. Rutgers- The State University, New Jersey (1969)
68. Taylor, W.A.: Quick switching systems. *J. Qual. Technol.* **28**(4), 460–472 (1996)
69. Soundararajan, V., Arumainayagam, S.D.: Construction and selection of modified quick switching systems. *J. Appl. Statist.* **17**(1), 83–114 (1990)
70. Govindaraju, K.: Procedures and tables for selection of zero acceptance number quick switching system for compliance sampling. *Commun. Stat.- Simul. C.* **20**(1), 151–171 (1991)
71. Soundararajan, V., Arumainayagam, S.D.: Quick switching system for costly and destructive testing. *Sankhyā: Ind. J. of Stat. Ser. B* **54**(1), 1–12 (1992)
72. Govindaraju, K., Kuralmani, V.: Modified tables for the selection of QSS-1 quick switching system for a given (AQL,LQL). *Commun. Stat.- Simul. C.* **21**(4), 1103–1123 (1992)
73. Govindaraju, K.: Zero acceptance number chained quick switching system. *Commun. Stat.- Theor. M* **40**(12), 2104–2116 (2011)
74. Schilling, E.G., Neubauer, D.V.: *Acceptance Sampling in Quality Control*. Chapman and Hall/CRC, New York (2017)
75. Calvin, T.W.: TNT zero acceptance number sampling. In: *ASQC Technical Conference Transaction*, pp. 35–39. American Society for Quality Control, Philadelphia, PA (1977)
76. Soundararajan, V., Vijayaraghavan, R.: Construction and selection of Tightened-Normal-Tightened (TNT) plans. *J. Qual. Technol.* **22**(2), 146–153 (1990)
77. Subramani, K., Govindaraju, K.: Selection of zero acceptance number Tightened-Normal-Tightened scheme for given (AQL, LQL). *Int. J. Manag. Sys.* **10**, 13–20 (1994)
78. MIL-STD-105 D: *Sampling Procedures and Tables for Inspection by Attributes*. US Government Printing Office. US Department of Defense, Washington DC (1963)
79. ISO 2859-1: *Sampling Procedures for Inspection by Attributes—Part 1: Sampling plans indexed by Acceptable Quality Level (AQL) for lot-by-lot inspection*. International Standards Organization, Geneva (1999)
80. Soundararajan, V., Vijayaraghavan, R.: Construction and selection of tightened-normal-tightened sampling inspection scheme of type TNT-($n_1, n_2; c$). *J. Appl. Statist.* **19**(3), 339–349 (1992)
81. Dodge, H.F.: Skip-Lot sampling plan. *Indust. Qual. Contr.* **11**, 3–5 (1955)
82. Perry, R.L.: Skip-Lot sampling plans. *J. Qual. Technol.* **5**(3), 123–130 (1973)

83. Govindaraju, K.: Contributions to the study of certain special purpose plans Ph.D. Thesis. University of Madras, Chennai (1985)
84. Govindaraju, K., Balamurali, S.: Chain sampling plan for variables inspection. *J. Appl. Statist.* **25**(1), 103–109 (1998)
85. Govindaraju, K., Kuralmani, V.: A note on the operating characteristic curve of the known sigma single sampling variables plan. *Commun. Stat. Theor. M.* **21**(8), 2339–2347 (1992)
86. Luca, S.: Modified chain sampling plans for lot inspection by variables and attributes. *J. Appl. Statist.* **45**(8), 1447–1464 (2018)
87. Balamurali, S., Jun, C.-H.: Multiple dependent state sampling plans for lot acceptance based on measurement data. *Eur. J. Oper. Res.* **180**(3), 1221–1230 (2007)
88. Balamurali, S., Jeyadurga, P., Usha, M.: Optimal designing of a multiple deferred state sampling plan for Weibull distributed life time assuring mean life. *Am. J. Math. Manag. Sci.* **36**(2), 150–161 (2017)
89. Balamurali, S., Jeyadurga, P., Usha, M.: Optimal designing of multiple deferred state sampling plan for assuring percentile life under Weibull distribution. *Int. J. Adv. Manufac. Tech.* **93**(9), 3095–3109 (Dec 2017)
90. Balamurali, S., Jeyadurga, P., Usha, M.: Determination of optimal quick switching system with varying sample size for assuring mean life under Weibull distribution. *Sequen. Anal.* **37**(2), 222–234 (2018)
91. Aslam, M., Azam, M., Jun, C.-H.: Multiple dependent state repetitive group sampling plan for Burr XII distribution. *Qual. Eng.* **28**(2), 231–237 (2016)
92. Arizono, I., Kanagawa, A., Ohta, H., Watakabe, K., Tateishi, K.: Variable sampling plans for normal distribution indexed by Taguchi's loss function. *Naval Res. Logist. (NRL)* **44**(6), 591–603 (1997)
93. Wu, C.-W., Lee, A.H.I., Chen, Y.-W.: A novel lot sentencing method by variables inspection considering multiple dependent state. *Qual. Reliab. Eng. Int.* **32**(3), 985–994 (2016)
94. Wu, C.-W., Wang, Z.-H.: Developing a variables multiple dependent state sampling plan with simultaneous consideration of process yield and quality loss. *Int. J. Prod. Res.* **55**(8), 2351–2364 (2017)
95. Wu, C.-W., Lee, A.H.I., Chien, C.-C.C.: A variables multiple dependent state sampling plan based on a one-sided capability index. *Qual. Eng.* **29**(4), 719–729 (2017)
96. Wu, C.-W., Lee, A.H., Liu, S.-W., Shih, M.-H.: Capability-based quick switching sampling system for lot disposition. *Appl. Math. Model.* **52**, 131–144 (2017)
97. Kurniati, N., Yeh, R.-H., Wu, C.-W.: Designing a variables two-plan sampling system of type TNTVSS- $(n_T, n_N; k)$ for controlling process fraction nonconforming with unilateral specification limit. *Int. J. Prod. Res.* **53**(7), 2011–2025 (2015)
98. Yan, A., Liu, S., Dong, X.: Designing a multiple dependent state sampling plan based on the coefficient of variation. *SpringerPlus* **5**(1), 1447 (2016)
99. Page, E.S.: Continuous inspection schemes. *Biometrika* **41**(1/2), 100–115 (1954)
100. Page, E.S.: Cumulative sum charts. *Technometrics* **3**(1), 1–9 (1961)
101. Qiu, P.: *Introduction to Statistical Process Control*. Chapman and Hall/CRC, New York (2013)
102. Beattie, D.W.: A continuous acceptance sampling procedure based upon a cumulative sum chart for the number of defectives. *J. R. Statist. Soc. Ser. C (Appl. Statist.)* **11**(3), 137–147 (1962)
103. Prairie, R.R., Zimmer, W.J.: Graphs, Tables and discussion to aid in the design and evaluation of an acceptance sampling procedure based on cumulative sums. *J. Qual. Technol.* **5**(2), 58–66 (1973)
104. Elder, R.S., Provost, L.P., Ecker, O.M.: United States Department of Agriculture CUSUM acceptance sampling procedures. *J. Qual. Technol.* **13**(1), 59–64 (1981)
105. Beattie, D.W.: Patrol inspection. *J. R. Statist. Soc. Ser. C (Appl. Statist.)* **17**(1), 1–16 (1968)
106. Lucas, J.M.: Control schemes for low count levels. *J. Qual. Technol.* **21**(3), 199–201 (1989)
107. Govindaraju, K., Lai, C.D.: Statistical design of control schemes for low fraction nonconforming. *Qual. Eng.* **11**(1), 15–19 (1998)
108. Lucas, J.M., Crosier, R.B.: Fast initial response for CUSUM quality-control schemes: give your CUSUM a head start. *Technometrics* **42**(1), 102–107 (2000)
109. Gao, Y., Tang, L.-C.: The effect of correlation on chain sampling plans. *Qual. Reliab. Eng. Int.* **21**(1), 51–61 (2005)
110. Baker, W.E., Thomas, J.: Armor acceptance procedure. *Qual. Eng.* **5**(2), 213–223 (1992)
111. Suresh, K.K., Devaarul, S.: Designing and selection of mixed sampling plan with chain sampling as attribute plan. *Qual. Eng.* **15**(1), 155–160 (2002)
112. Suresh, K.K., Devaarul, S.: Multidimensional mixed sampling plans. *Qual. Eng.* **16**(2), 233–237 (2003)
113. Aslam, M., Balamurali, S., Azam, M., Rao, G., Jun, C.: Mixed multiple dependent state sampling plans based on process capability index. *J. Test. Eval.* **43**, 171–178 (2015)
114. Balamurali, S.: A new mixed chain sampling plan based on the process capability index for product acceptance. *Commun. Stat.-Simul. C.* **46**(7), 5423–5439 (2017)
115. Usha, M., Balamurali, S.: Designing of a mixed-chain sampling plan based on the process capability index with chain sampling as the attributes plan. *Commun. Stat. Theor. M* **46**(21), 10456–10475 (2017)
116. Balamurali, S., Usha, M.: Designing of variables modified chain sampling plan based on the process capability index with unknown mean and variance. *Am. J. Math. Manag. Sci.* **36**(4), 363–377 (2017)
117. Balamurali, S., Aslam, M., Liaquat, A.: Determination of a new mixed variable lot-size multiple dependent state sampling plan based on the process capability index. *Commun. Stat. Theor. M* **47**(3), 615–627 (2018)
118. Balamurali, S., Usha, M.: Optimal designing of variables quick switching system based on the process capability index C_{pk} . *J. Indust. Prod. Eng.* **31**(2), 85–94 (2014)
119. Bray, D.F., Lyon, D.A., Burr, I.W.: Three class attributes plans in acceptance sampling. *Technometrics* **15**(3), 575–585 (1973)
120. Shankar, G., Mohapatra, B.N., Joseph, S.: Chain sampling plan for three attribute classes. *Int. J. Qual. Relia. Manag.* **8**, 46–55 (1991)
121. Suresh, K.K., Deepa, O.S.: Risk based Bayesian chain sampling plan. *Far East J. Theor. Statist.* **8**, 121–128 (2002)
122. Latha, M., Jeyabharathi, S.: Bayesian chain sampling plan based on binomial distribution. *J. Stat. Manag. Sys.* **16**(6), 391–400 (2013)
123. Arul, S.D., Moses, V.: Design and development of relational chain sampling plans RChSP(0, i) for attribute quality characteristics. *Cogent Math.* **4**(1), 1408232 (2017)
124. Tang, L.-C., Cheong, W.-T.: A control scheme for high-yield correlated production under group inspection. *J. Qual. Technol.* **38**(1), 45–55 (2006)
125. Aslam, M., Arif, O.H., Jun, C.-H.: A new variable sample size control chart using MDS sampling. *J. Statist. Comp. Simul.* **86**(18), 3620–3628 (2016)
126. Arshad, A., Azam, M., Aslam, M., Jun, C.-H.: A control chart for monitoring process variation using multiple dependent state sampling. *Commun. Stat. Simul. C.* **47**(8), 2216–2233 (2018)
127. Stephens, K.S.: *The Handbook of Applied Acceptance Sampling: Plans, Principles, and Procedures*. ASQ Quality Press, Mexico (2001)



Dr. Govindaraju Kondaswamy is an academic staff in the Statistics and Bioinformatics Group within the School of Fundamental Sciences, Massey University, Palmerston North, New Zealand. He holds a PhD degree in Statistics. His research specialisation includes acceptance sampling, SPC, industrial experiments and data analysis. He has keen interest in developing Shiny apps and R packages for practitioners.



Contents

13.1	Introduction	240
13.1.1	What Is Six Sigma?.....	240
13.1.2	Why Six Sigma?.....	241
13.1.3	Six Sigma Implementation.....	241
13.2	The DMAIC Methodology	242
13.2.1	Introduction.....	242
13.2.2	The DMAIC Process.....	242
13.2.3	Key Tools to Support the DMAIC Process.....	244
13.3	Design for Six Sigma	246
13.3.1	Introduction.....	246
13.3.2	Why DFSS?.....	246
13.3.3	Design for Six Sigma: The DMADV Process.....	247
13.3.4	Key Tools to Support the DMADV Process.....	248
13.4	Six Sigma Case Study	252
13.4.1	Introduction.....	252
13.4.2	Process Background.....	252
13.4.3	Define Phase.....	253
13.4.4	Measure Phase.....	253
13.4.5	Analyze Phase.....	253
13.4.6	Improve Phase.....	253
13.4.7	Control Phase.....	254
13.5	Six Sigma and Big Data	254
13.5.1	Background.....	254
13.5.2	DMAIC: New Opportunities.....	256
13.6	Conclusion	258
References	258

Abstract

Six Sigma, which was first launched by Motorola in the late 1980s, has become a successful standard-quality initiative to achieve and maintain excellent business performance in today's manufacturing and service industries. In this chapter, we provide a systematic and principled introduction of Six Sigma from its various facets. The first part of this chapter describes what Six Sigma is, why we need Six Sigma, and how to implement Six Sigma in practice. A typical business structure for Six Sigma implementation is introduced, and potential failure modes of Six Sigma are also discussed.

The second part describes the core methodology of Six Sigma, which consists of five phases, i.e., Define, Measure, Analyze, Improve, and Control (DMAIC). Specific operational steps in each phase are described in sequence. Key tools to support the DMAIC process including both statistical tools and management tools are also presented. The third part highlights a specific Six Sigma technique for product development and service design, Design for Six Sigma (DFSS), which is different from DMAIC. DFSS also has five phases: Define, Measure, Analyze, Design, and Verify (DMADV), spread over product development. Each phase is described, and the corresponding key tools to support each phase are presented.

In the fourth part, a real case study on printed circuit board (PCB) improvement is used to demonstrate the application of Six Sigma. The company and process background are provided. The DMAIC approach is specifically followed, and key supporting tools are illustrated accordingly. At the end, the financial benefit of this case is realized through the reduction of cost of poor quality (COPQ). The fifth part provides a discussion of Six Sigma in current Big Data background. A brief introduction of Big Data is first given, and then the tremendous opportunities offered by Big Data analytics to the core methodology of Six Sigma, i.e., DMAIC, are outlined in detail. The capabilities of each phase that would be greatly enhanced are

F. Tsung (✉)
Department of Industrial Engineering and Decision Analytics, Hong Kong University of Science and Technology, Hong Kong, China
e-mail: season@ust.hk

K. Wang
School of Management, Xi'an Jiaotong University, Xi'an, Shaanxi, China
e-mail: kwangai@xjtu.edu.cn

emphasized. Finally, the last part is given to conclusions and a discussion of prospects of Six Sigma.

Keywords

DFSS · DMAIC · DOE · Quality Big Data · Quality engineering · SPC · Six Sigma

13.1 Introduction

Since the early 1990s, Six Sigma has been sweeping the business world, driving an unprecedented emphasis on greater manufacturing and service quality. Six Sigma is one of the few quality initiatives that actually originated from industrial practice. Six Sigma was originally devised as a measure of quality that strives for near perfection. It has developed into a disciplined, data-driven, and customer-focused approach to reduce defects and bring about substantial financial growth. Although most of the Six Sigma efforts were focused on manufacturing operations in the early years, the Six Sigma approach has now been more widely used in the nonmanufacturing industry, such as in finance, insurance, health care, and telecommunications. Users include American Express, American International Group (AIG), Bank of America, Citibank, JPMorgan, Chase, Merrill Lynch, Vanguard, etc. These companies have actually seen larger business impacts and cost savings of Six Sigma than those in manufacturing.

13.1.1 What Is Six Sigma?

Motorola first introduced the Six Sigma program in the late 1980s with the aim to increase profitability by reducing defects. General Electric (GE) followed the approach at their manufacturing sites and later at their financial service divisions. After that, Six Sigma is thought to be applicable to all processes and transactions within GE. Six Sigma has now evolved from a quality improvement program to an overall business strategy executive system and business results-oriented program, which seems more “total” than “total quality management (TQM).” We describe the basic definition of Six Sigma in this section and elaborate its systematic methodology and business structure in later sections.

Six Sigma is both a business improvement strategy and a methodology to measure process performance. It is used to increase profits by eliminating defects, waste, and variability, and to find the causes of mistakes in products, processes, and services to increase yields. In Six Sigma, a focus on the

customer is the top priority. Performance standards are based on actual customer input, so that process effectiveness can be measured and customer satisfaction can be predicted.

In terms of business process improvement, variation reduction is the key since variation signals fluctuation in the process output and, often times, is a major source of poor quality. **Variation** is present in all processes and every aspect of work. Unintended variation reduces process performance and decreases customer satisfaction. Because of the existence of variation, producing high-quality products and services in the modern industrial environment is a tough task.

Therefore, Six Sigma aims particularly at reducing variation. The word “sigma” or the symbol “ σ ” is used in statistical notation to represent the standard deviation in a population. The standard deviation is used as a general measure of variation in any kind of product or process. With six standard deviations between the process mean and the customer’s specification limit, we arrive at 3.4 **defects per million opportunities (DPMO)**, that is, a 99.9997 percent yield. Before the Six Sigma technique was introduced, a three-sigma level of variation was regarded as being fairly good-quality performance. Three Sigma may be acceptable for a product or process having only a single or a few stages. It is not good enough for many products that are the result of hundreds of thousands of stages, such as automobiles and computers.

For example, if a production process is made up of ten stages where the yield of each stage is as high as 90%, the probability of producing a satisfactory product in the final run would be $0.9^{10} = 35\%$. This indicates that about 65% of the products are defective. If a production process is made up of 100 stages, the probability of producing a satisfactory product under the **Three Sigma** program could be as low as 0.1%, as shown in Table 13.1. The Six Sigma regime, however, allows only 3.4 defects for every million opportunities, which ensures a quality product even if the process involves a large number of stages (see Table 13.1). Part of the reason for using such a strict requirement in quality management is actually to accommodate the common multistage processes in modern industrial practice.

Table 13.1 Final yield for different sigma levels in multistage processes

Average sigma level	1	2	3	4	5	6
Final yield for 10 stages	0.0%	2.5%	50.1%	94.0%	99.8%	100.0%
Final yield for 100 stages	0.0%	0.0%	0.1%	53.6%	97.7%	100.0%
Final yield for 1000 stages	0.0%	0.0%	0.0%	0.2%	79.2%	99.7%

13.1.2 Why Six Sigma?

The successful implementation of Six Sigma can benefit the areas of cost reduction, increased profit, increased market share, and enhanced business competitiveness, mainly by the reduction of the **cost of poor quality (COPQ)**.

COPQ usually includes appraisal costs, internal failure costs, and external failure costs. Appraisal and inspection costs are often incurred, for example, in checking finished goods before they leave the factory, inspecting purchased equipment/supplies, proofreading financial and legal documents, reviewing charges prior to billing, etc. Internal failure costs are those for repairing, replacing, or discarding work in progress or completed work before the delivery of the product to the customer. External failure costs are those that directly affect the customer and are the most expensive to correct, including the tangible remedial costs and the intangible costs associated with losing dissatisfied customers.

COPQ cannot be underestimated. In manufacturing industries, COPQ sometimes reaches 15% of total sales (source: **Six Sigma Academy**). In service industries, the situation is even more serious. COPQ may account for as much as 50% of total costs. However, these COPQ could be saved with the use of Six Sigma. General Electric has estimated saving of 2 billion US dollars during the first 5 years of Six Sigma implementation, and Allied Signal has estimated saving of 1.1 billion US dollars in 2 years. Indeed, thousands of companies around the world have enjoyed the breakthrough benefits of Six Sigma. For example, Legend Computers in China reported in 2002 savings of \$20 million dollars during the first year of implementation. In the same year, the International Bank of Asia in Hong Kong reported savings of 1.4% of total costs during the first year of Six Sigma implementation.

13.1.3 Six Sigma Implementation

Six Sigma implementation is usually a top-down approach, i.e., from the strong commitment of top management. As most Six Sigma projects span across several departments, the organizational barrier could not be removed without leadership commitment to Six Sigma. Strong commitment, leadership, and strategic involvement have proven to be key factors of Six Sigma's success. Second, as Six Sigma requires a long-term mentality, it needs to be positioned first as a strategic initiative and then be linked to operational goal. It is important to tie the Six Sigma implementation to corporate goals, such as increased profits through lower costs and higher loyalty as an example. Also, effective internal communication is another key issue for the success of Six Sigma implementation.

In the following, a typical business structure for Six Sigma implementation is introduced. Several potential failure modes and practical considerations of Six Sigma implementation are also discussed.

Training and Belt Structure

The deployment of Six Sigma in a company usually starts with education. Without necessary training, people are not able to bring about the breakthrough improvement of Six Sigma. Six Sigma establishes well-defined and structural roles and responsibilities for a project team, and team members are given formal training according to their roles to help the team work effectively. A Six Sigma team is usually organized in a **belt structure** (as in martial arts) as follows.

At the top of the belt structure is the **Six Sigma Executive**. The Six Sigma Executive could be a council that consists of top managers who have the vision and make strategic decisions for a company. They are responsible for establishing the roles and structures of Six Sigma projects. They also need to make decisions on project selection and resources allocations. A progress review is conducted periodically to monitor projects.

Champions are the senior managers who supervise Six Sigma projects. They report directly to the Six Sigma Executive and represent the team to the Executive. They also need to seek resources and to learn the focus of the business from the Executive. In addition, Champions meet Black Belts and Green Belts periodically to review the progress and coach the team.

Master Black Belts work with the Champions to ensure that Six Sigma objectives and targets are set. Meanwhile, they are the statistical problem-solving experts in a company. Their responsibilities include determining plans, providing technical expertise, and training and coaching Black and Green belts.

Black Belts, as on-site Six Sigma experts, usually possess the technical background needed to help Green belts and the other team members to understand the project and apply appropriate statistical techniques. Their roles are to provide formal training to local personnel in new strategies and tools, provide one-on-one support to local personnel, pass on new strategies and tools in the form of training, workshops, case studies, local symposia, etc., and find application opportunities for breakthrough strategies and tools, both internal and external (i.e., to the suppliers and customers).

Green Belts, on the other hand, execute Six Sigma in their specific area as a part of their overall job. They may assist Black belts in completing sections of their projects and apply their learning in their daily performance of their jobs.

According to the Six Sigma Academy, Black Belts are able to save companies approximately US\$230,000 per project and can complete four to six projects per year. The American Society for Quality (ASQ) provides the certification of Six Sigma Black Belts (SSBB) internationally.

Six Sigma Failures (Sick Sigma)

Although Six Sigma is a powerful approach, it can lead to failure when some critical issues are neglected. However, as more companies have implemented Six Sigma since the 1990s, the factors that have led to failure have been identified and summarized. According to [1], project selection and management support are usually the two main sources of failure.

The **failure modes** in project selection usually include projects not tied to financial results, poorly defined project scopes, metrics and goals, projects lasting more than 6 months, the wrong people assigned to projects, project teams that are too large, and infrequent team meetings. On the other hand, the failure modes in management support may include Black belts with little time to work on projects, poor or infrequent management reviews, poor support from finance, information technology (IT), human resource (HR), and poor communication of initiatives and progress [1].

Especially, for a Six Sigma program to sustain without failure, **recognition and reward systems** would be the key. If a lack of or no change on the recognition and reward systems, the program has no way to last. Necessary practices include establishing and using selection and promotion criteria and developing corresponding performance management and reward systems. GE way which links 40% of management bonus to Six Sigma may be too aggressive, but a company must adequately compensate those high-performing members.

Note that the use of statistical methods is not on the list of major failure modes. With the recent advances in information technology, computing and sensing technology, the use of advanced statistical methods has become handy via commercial software packages (such as MINITAB, JMP, etc.). Therefore, the use of statistical tools is no longer a bottleneck in Six Sigma implementation.

Besides, various industry types and company natures are also not an excuse for the Six Sigma failure. Six Sigma has been successfully applied to many processes outside of manufacturing, regardless of their company size or their industry nature. In particular, transactional processes, such as software coding, billing, customer support, etc., often contain variation or excessive cycle time and can be optimized by applying Six Sigma. For example, HR managers may apply it to reduce the cycle time for hiring employees, and regional sellers may apply it to improve forecast reliability, pricing strategies, or variations.

13.2 The DMAIC Methodology

13.2.1 Introduction

The development of Six Sigma is evolutionary, not revolutionary, and it integrates many useful quality management tools. Thus, it is not surprising to find overlaps between Six Sigma, TQM, Lean, and ISO. The core methodology of Six Sigma is driven by the close understanding of customers' needs and the disciplined use of facts, data, and statistical analysis, which consists of five phases, i.e., Define, Measure, Analyze, Improve, and Control (DMAIC).

In the Define phase, the specific problem is identified, and the project goals and deliverables are defined. In the Measure phase, the critical-to-quality (CTQ) characteristics are identified and the measurement system is reviewed. The nature and properties of the data collection have to be understood thoroughly to ensure the quality of the data. In the Analyze phase, both quantitative (i.e., statistical) methods and qualitative (i.e., management) tools are used to isolate the key information that is important to explaining defects. In the Improve phase, the key factors that cause the problem should be discovered. In the Control phase, the key factors and processes are controlled and monitored continuously to ensure that the improvement is sustainable and the problem will not occur again. A detailed case study on the implementation of the DMAIC methodology in printed circuit board (PCB) manufacturing can be found in [2]. The paper "Six Sigma approach to reducing fall hazards among cargo handlers working on top of cargo containers: a case study" [3] is another case study using DMAIC that focuses on a nonmanufacturing case.

13.2.2 The DMAIC Process

More specifically, we implement the DMAIC methodology in detailed steps in sequence in order to shift our focus from the output performance (i.e., y) to the root causes (i.e., the x). Based on these steps, we transfer a practical problem into a statistical problem (e.g., mapping x and y), find out a statistical solution for that (e.g., solving $y = f(x)$) and then transfer the statistical solution back into a practical solution. Each step is described in the following, and the corresponding key tools will be further explained in a later section.

Phase 1. Define (D)

This phase defines the Six Sigma project that includes a problem statement, the objectives of the project, the expected benefits, the team structure, and the project time line. At the end of this phase, we should have a clear operational

definition of the project metrics for the final evaluation. In this phase, the main tasks are to identify who the customer is, select the project area, define the goal, scope, and resources of the project, form a Six Sigma project team, define the team members' responsibilities, and estimate the profit and cost for this project to ensure the value of the project. Key Tools in this phase include the project charter, business process mapping, SIPOC, etc.

Phase 2. Measure (M)

By taking steps in the measure phase, we have a clear understanding of the performance of the current process, and only after knowing where we are now, we can determine where we should be in the future. Three implementation steps in this phase are to select the critical-to-quality (CTQ) measures, determine deliverables, and quantify the measurability of y .

Select the Critical to Quality (CTQ) Measures

In this step, we will identify the external CTQ from the customer point of view (i.e., the big Y) that will be improved, and then link that with the internal CTQ (i.e., the small y), which is a quantitative measure in the company and will be the focus of the project. Key tools in this step include Customer Needs Mapping (CNM), Quality Function Deployment (QFD), Failure Modes and Effects Analysis (FMEA), etc.

Determine Deliverables

We establish a performance standard and develop a data collection plan for the internal CTQ y in this step. If the measure of y from the previous step is attributal, what is the definition of a defect? If the data are continuous, what are the lower and upper specifications for defectiveness? Key tools used in this step include process mapping and yield calculation.

Quantify Measurability

We validate the measurement system on how to ensure the measurement results are accurate for the following analysis. We may need to improve the measurement system before continuing. Key tools include measurement system analysis (MSA), Gauge R&R study.

Phase 3. Analyze (A)

After we identify the “ y ” in the process, we need to determine the x (root causes), which may impact the performance of “ y .” In the analyze phase, we use various management and statistical tools to discover the x 's for future improvements. Three implementation steps in this phase are to establish the baseline, determine the improvement plan, and identify the sources of variation.

Establish the Baseline

We will establish the process capability for the current process to understand where we are now. We need to collect

the current process data, use graphical tools to analyze the data, and calculate the process capability indices, the defect per million opportunities (DPMO), and the sigma level (Z). Key tools include the following: histograms, process capability indices (PCI), etc.

Determine Improvement Plan

We quantify the goals for the improvement to make the aim of the project clear, and we may determine if the goal is significantly different from today's performance (i.e., the baseline) through hypothesis testing. Key tools include benchmarking, hypothesis testing, t test, ANOVA, etc.

Identify Variation Sources

We list all the potential factors (x 's) that may influence the performance of “ y .” Regression analysis may be conducted where applicable to identify potential x 's. Key tools include brainstorming, cause and effect diagram, regression analysis, etc.

Phase 4. Improve (I)

As the root causes for variation are obtained, it becomes possible for us to fix these root causes. In the improve phase, the way that we can achieve a better process needs to be found, where the Design of Experiments (DOE) is a key technique to help us quantify the relation between y and x 's, and to improve the process by finding the optimal setting of x 's for each y . In this phase, we follow three implementation steps: screen potential sources of variation, discover variable relationships, and formulate the implementation plan.

Screen Potential Sources of Variation

We determine the vital few x 's from the trivial many x 's in this step. DOE is a key tool for factor screening. Both full factorial and fractional factorial experiments can be used. If necessary, historical data can be used with care, and a similar model or simulation may be used as well.

Discover Variable Relationships

We develop the transfer function ($y = f(x)$) linking the y to the vital x 's. Based on that, we then determine and confirm the optimal settings for the vital few x 's. DOE is a key tool for characterization and optimization as well. Various DOE techniques, such as the response surface method (RSM), robust design, and the Taguchi method, can be applied in this step. Other than that, simulation or surveys can also be used to find the relationship.

Formulate Implementation Plan

In this step, if a new process or process steps have been put in place, show the new process map. For the new process, indicate the new in-process measurements and associated specifications. If there is not a new process, indicate any new measurements put in place. We list how the changes to the x 's will be implemented and how much flexibility is available in the settings of each x . Key tools in this step include tolerance design, main effects plots, and interaction plots.

Phase 5. Control (C)

After determining how to fix the process, we want the improvement for the process to be sustainable. The control phase is set up to ensure the sustainable improvement and to deploy measurement tools to confirm that the process is in control. It is also critical to realize the financial benefits and develop a transfer plan in this phase. Three implementation steps include validating the implementation plan, controlling the inputs and monitoring the outputs, and finally sustaining the change.

Validate the Implementation Plan

To determine how well the x 's can be controlled, we will validate the measurement system on the x 's, and we may need to improve measurement system before continuing. We will also report new sigma levels and new DPMO levels at this step. Key tools include Gauge R&R, ANOVA, etc.

Control Inputs and Monitor Outputs

We determine how each vital x can be controlled (e.g., attribute control chart, variable control chart, mistake-proofing, etc.) and set up a monitoring plan for the y and x 's in this step. Key tools include statistical process control (SPC), attribute control charts, variable control charts, Poka-Yoke (mistake-proofing), etc.

Sustain the Change

The objective of this step is to ensure that changes last after the improvement strategy has been implemented. Process control plans need to be developed and implemented for each x . We will also verify financial gains can be achieved and if this project is translatable to any other regions, lines, sites, processes, etc. Key tools in the final step include out-of-control plans, mistake-proofing, audit strategy, etc.

13.2.3 Key Tools to Support the DMAIC Process

This section presents the key tools to support the DMAIC process. Only a few key tools can be covered in this section, and each method is outlined briefly with the basic ideas and mechanisms. The books and papers cited in this section give more details.

A. *Business Process Mapping (SIPOC Diagrams)*

Purpose:

SIPOC stands for Suppliers, Inputs, Process, Outputs, and Customer. SIPOC diagrams are graphical tools to identify all relevant elements or stakeholders of a business process and map the process flow before the project begins. They are usually used in the Define phase.

Definition:

Supplier: Whoever produces, provides, or furnishes the products or services for the input of the process, either an internal or an external supplier

Inputs: Material, resources, and data required to execute the process

Process: The manufacturing or service process of interest which transforms the inputs into the outputs

Outputs: The tangible products or intangible services that result from the process

Customer: Whoever receives the outputs of the process, either an internal customer or an external customer

How to do it:

Step 1. Clear statement of CTQ and the process.

Step 2. Clear statement of start/end point.

Step 3. Identify major customers, suppliers, outputs, and inputs.

Step 4. Identify the five to seven major process steps using brainstorming and storyboarding.

Step 5. Decide what step to map in detail.

Step 6. Complete detail map.

B. *Quality Function Deployment (QFD)*

QFD is a systematic approach to prioritize and translate customer requirements (i.e., external CTQ) into appropriate company requirements (i.e., internal CTQ) at each stage from product development to operations to sales/marketing to distribution. This method is usually used in the Measure phase. It is also useful in the design for Six Sigma (DFSS) and will be introduced in more detail in the DFSS section.

C. *Failure Modes and Effects Analysis (FMEA)*

Purpose:

FMEA is a tool to reduce the risk of failures. It is also a tool to identify and prioritize CTQ at the Measure phase.

Definition:

Severity: The assessment of how severe a failure mode is. The severity usually scales from 1 to 10. Scale 1 means a minor failure mode that may not be noticed, and 10 means a very serious failure that may affect safe operations.

Occurrence: The likelihood that a specific cause will result in the failure mode, which scales from 1 to 10 with 10 the highest likelihood.

Detection: The assessment of the ability to identify the failure mode. A 1–10 scale is often used with 10 the lowest detectability.

RPN: The risk priority number (RPN) is the output of an FMEA. $RPN = \text{Severity} \times \text{Occurrence} \times \text{Detection}$.

How to do it (refer to [4, 5]):

- Step 1.** Identify the products, services, or processes.
- Step 2.** Identify the potential failure that would arise in the target process.
- Step 3.** Identify the causes of the effects and their likelihood of occurrence.
- Step 4.** Identify the current controls for detecting each failure mode and the ability of the organization to detect each failure mode.
- Step 5.** Calculate RPN by multiplying the values of severity, potential causes, and detection.
- Step 6.** Identify the action for reducing or eliminating the RPN for each failure mode.

D. Measurement System Analysis (MSA)

Purpose:

A statistical evaluation of the measurement system must be undertaken to ensure effective analysis of any subsequent data generated for a given process/product characteristic. MSA is usually used in the Measure and Control phases to validate the measurement system for y and x 's.

Definition:

- Gauge R&R:** A tool to study the variation in the measurement process arising from the measurement device and the people taking the measurement.
- Repeatability:** The variability that reflects the basic inherent precision of the gauge itself.
- Reproducibility:** The variability due to different operators using the gauge (or different time, different environments) [6].

How to do it:

- Step 1.** Collect the data. Generally, two to three operators, ten units to measure, and each unit is measured two-three times by each operator.
- Step 2.** Perform the calculations to obtain %R&R (see [6]).
- Step 3.** Analyze the results. A rule of thumb is that:
 - %R&R < 10%: measurement system is acceptable.
 - %R&R between 10% and 30%: measurement system may be acceptable. We will make decisions based on the classification of the characteristics, hard applications, customer inputs, and the Sigma levels of the process.
 - %R&R > 30%: measurement system is not acceptable. We should improve the measurement system by finding problems and removing root causes.

E. Process Capability Analysis

Purpose:

Process capability analysis is a statistical technique to quantify process variability, analyze this variability relative to customer requirements or specifications, and assist in reducing the variability [6]. It is used in the Analyze phase.

Definition:

- Cp:** Process/Product capability index is the relationship of the process/product variation to the upper and lower specification limits. It is related to the potential process capability and not a measure of how centered the data are.
- Cpk:** It compares process variability with the specification's width and location. It takes into account that the sample mean may be shifted from the target. Since both the mean shift and the variability of the characteristics are considered, Cpk is better related to the capability of the current process.

How to do it:

The detailed calculation and analysis are given in Chap. ▶ 8 in [6].

F. Cause-Effect Diagram (Fishbone Diagram)

Purpose:

This is a graphical brainstorming tool to explore the potential causes (i.e., x 's) that result in a significant effect on y . It is usually used in the Analyze phase.

How to do it:

- Step 1.** Define clearly the effect or analyzed symptom (y) for which the possible causes (x 's) must be identified.
- Step 2.** Place the effect or symptom (y) being explained on the right of a sheet of paper.
- Step 3.** Use brainstorming or a rational step-by-step approach to identify the possible causes in areas related to man, machine, material, method, and environment.
- Step 4.** Each of the major areas of possible causes should be connected with the central spine by a line.
- Step 5.** Add possible causes (x 's) for each main area.
- Step 6.** Check for completeness.

G. Design of Experiments (DOE)

Purpose:

DOE is a major tool in the Improve phase. It is used for screening the vital few x 's, characterizing the relationship between y and x 's, and optimizing the setting of the vital x 's.

Definition:

Factor: An independent variable (i.e., x) whose state can be varied.

Level of a factor: The state of the factor.

Full Factorial Experiments: Discover the factor effects and relationship between y and x 's by running all the combinations of factor levels.

Fractional Factorial Experiments: An economical approach to discovering the factor effects and to screening the vital few x 's by running only part of the combinations of factor levels.

Response Surface Methodology (RSM): A DOE technique that is useful for modeling and optimization in which a response of interest y is influenced by several factors x 's and the objective is to optimize this response. This method is discussed more in the DFSS section.

How to do it (see [7, 8]):

Step 1. State the problem.

Step 2. Choose the response variable (y).

Step 3. Choose the factors (x 's) and their levels and ranges.

Step 4. Determine the experimental plan (i.e., the design matrix).

- (a) For screening the x 's to obtain vital few x 's, we often use factorial experiments. In such cases, if the number of runs is moderate and we have enough time and resources, we may conduct a full factorial experiment; if the number of runs is large or time and resources are limited, we may consider a fractional factorial experiment.
- (b) For obtaining the optimal response, we may conduct RSM, which is usually conducted after variable screening.

Step 5. Run the experiments under the prescribed conditions and collect the response data.

Step 6. Analyze the data collected using main effect plots, interaction plots, ANOVA, etc.

Step 7. Conclude the experiment and make recommendations. A confirmation run or a follow-up DOE is usually needed.

H. *Statistical Process Control (SPC)**Purpose:*

SPC is a major tool in the Control phase. It is used to control and monitor the stability and capability of the vital few x 's for CTQ.

How to do it:

This method is discussed in more detail in the DFSS section. For a general introduction about SPC, see

[6, 9]. For recent advances in SPC, the readers may refer to <http://qlab.ieda.ust.hk> and references therein.

As a final note, we summarize the main purposes and key tools of each of the five phases demonstrated throughout Sect. 2 in the following figure, so that practitioners can have a better idea on how to systematically implement the DMAIC approach to solve a real quality problem (Fig. 13.1).

13.3 Design for Six Sigma

13.3.1 Introduction

The success of Six Sigma's DMAIC methodology has generated enormous interest in the business world. One of the basic ideas is to measure existing defective processes quantitatively and then to improve them. Compared with this defect-correction methodology, Design for Six Sigma (DFSS) is a proactive methodology, which focuses on the new product/service development to prevent quality defects the first time instead of solving problems when they happen in existing processes.

DFSS is a disciplined and statistical approach to product and service design that ensures the new designs can meet customer requirements at launch. The objective of DFSS is to eliminate and reduce the design vulnerabilities in both the conceptual and operational phases by employing scientific tools and statistical methods.

Unlike the DMAIC methodology, the phases of DFSS are not universally defined. There are many methodologies, such as Woodford's IDOV (Identify, Design, Optimize, Validate), El-haik's ICOV (Identify, Characterize, Optimize, Verify), Tennant's DCCDI (Define, Customer Concept, Design, and Implement), and so on. All these approaches share common themes, objectives, and tools. In this section, we refer to above methodologies, especially General Electric's DFSS approach called DMADV:

Define the project goals and customer requirements.

Measure and determine customer needs and specifications.

Analyze the options of conceptual solutions to meet customer needs.

Design the product/service to meet customer needs.

Verify the design performance and ability to meet customer needs.

13.3.2 Why DFSS?

Proactive Versus Retroactive

During the product/service design process, conceiving, evaluating, and selecting good design solutions are difficult tasks

Define	Identify project, champion and project owner Determine customer requirements and CTQs Define problem, objective, goals and benefits Define stakeholder/resource analysis Map the process Develop project plan	Key analytical tools Process mapping
Measure	Determine critical Xs and Ys Determine operational definitions Establish performance standards Develop data collection and sampling plan Validate the measurements Measurement system analysis Determine process capability and baseline	Measurement system analysis & process capability, QFD, pareto diagram, check sheet
Analyze	Benchmark the process or product Establish causal relationships using data Analysis of the process map Determine root causes using data	Statistical tests, cause-effect diagram, histogram, scatter plot, modeling & root cause analysis
Improve	Develop solution alternatives Assess risks and benefits of solution alternatives Validate solution using a pilot Implement solution Determine solution effectiveness using data	Brainstorming, design of experiments, FMEA, & validation
Control	Determine needed controls (measurement, design, etc.) Implement and validate controls Develop transfer plan Realize benefits of implementing solutions Close project and communicate results	Statistical process control, poka yoke

Fig. 13.1 Six sigma DMAIC road map

with enormous consequences. Usually, organizations operate in two modes: “proactive,” that is, conceiving feasible and healthy conceptual solutions the first time; and “retroactive,” that is, an after-the-fact practice that drives design in a design-test-fix-retest cycle and creates what is broadly known as the “fire fighting” mode of design. If a company follows this practice, it would suffer from high development costs, longer times to market, lower quality levels, and marginal competitive edge [10].

Compared to retroactive approaches such as DMAIC, which apply performance improvement in the later stages of the product/service life cycle, DFSS shifts the attention to improving performance in the front-end design stages. That is, the focus is on problem prevention instead of problem solving. This action is motivated by the fact that the design decisions made during the early stages of the product/service life cycle have the largest impact on both total cost and quality of the system. It is often claimed that up to 80% of the total cost is committed in the concept development stage. Also, at least 80% of the design quality is committed in the early design stages. According to a study of the design community [10], at the early design stage, the impact (influence) of design activity is much higher than a later stage, while the correction cost in the early stage is much lower.

Experience Dependency Versus Scientific and Systematic Methodology

Currently, most design methods are empirical in nature, while the work of the design community is often based on experience. This experience-based tradition often leads to unnecessary variation and is difficult for project manager to control. As a result, vulnerabilities are introduced to the new design that makes it impossible for the product/service to achieve Six Sigma performance. This is another motivation for devising DFSS as a scientific and systematic design method to address such needs.

13.3.3 Design for Six Sigma: The DMADV Process

Generally speaking, DFSS has five phases spread over product development. They are called DMADV, Define, Measure, Analyze, Design, and Verify.

Phase 1. Define (D)

The process of product/service design begins when there is a need (internal or external), which can be a problem to be solved or a new invention. In this phase, design objectives,

scope, and available resources should be simply and clearly defined in the design project charter as the key deliverables.

Phase 2. Measure (M)

In particular, the Voice of Customer (VOC) is the critical input in customer-oriented design. Based on the VOC, the internal CTQ measures (critical to quality or critical to satisfaction, i.e., the y), such as cost, performance, reliability, esthetics, and serviceability, need to be identified quantitatively and to be prioritized according to their importance to customers. This kind of information can help to define the function requirements in a later phase.

Phase 3. Analyze (A)

In this phase, the CTQs will be decomposed into measurable and solution-free Functional Requirements (FRs). Then, a number of conceptual-level design alternatives should be produced by the design team for the FRs, considering cost, physical properties, the difficulties to operate, manufacture, and maintain, etc. Through summarizing the design requirements and conceptual-level design alternatives, an overall set that contains high-potential and feasible solutions can be produced to help the design team to decide on the best solution considering the original design charter including the performance, the constraint of cost, and available resources.

Phase 4. Design (D)

Once the design team fixes the selection of the conceptual solutions, they need to decompose the FRs into Design Parameters (DPs). And at the same time, they need to consider the potential risk to achieve CTQs when they create detailed designs to the level of design parameters. Then, optimization tools will be used to get optimal values for the design parameters. In DFSS, optimization can be reached statistically, and by using statistical tools, the transfer functions can be generated to mathematically represent the relationships between the input and output of a product or a service process. Then, the design team can rely on the transfer function to optimize the design solutions so that the product/service can achieve a target performance and be insensitive to uncontrollable factors (noise factors), such as the environment and production case-to-case variation.

Phase 5. Verify (V)

In this phase, the design team makes a model formed by the simulation of a service process or a physical prototype that is the first working version of the product. Based on these few prototypes, the design team evaluates and tests the whole design to predict if future product's performance can meet the design charter and how to improve the solution when failure occurs.

13.3.4 Key Tools to Support the DMADV Process

Below is a summary of the key tools used to support the DMADV process.

A. Voice of Customer (VOC)

Purpose:

Define customer needs/requirements for the new product/service design or existing product/service redesign.

Input:

Market segment defined – who the customers are and their environment

Output:

Detailed customer requirements

How to do it (refer to [11]):

Step 1. Define market segments – to understand who the customers are and where to go to gather their needs.

Step 2. Identify objective for interviews of customer – to learn which of their needs are new, unique, and difficult (NUD).

Step 3. Select customer groups within the target market segments.

Step 4. Decide on the customer visit team – divide into three roles: leader, subordinate interviewer that helps adding balance and diversity in the discussion, and statement writer that writes down the VOC needs statement.

Step 5. Create an interview guide based on objectives – to get customers' responses that are rich in description of needs.

Step 6. Listen, probe, and observe customers by asking stimulating questions and open-ended statements to gather the VOC. Image data can be gathered by actual observation of customers' responses to existing products or services.

B. KJ Method (See [12])

Purpose:

Structure and rank the customer requirements

Input:

The detailed VOC

Output:

Organized customer requirements

How to do it (refer to [13]):

- Step 1.** Write down customer descriptions as “statements of customer requirements” on a POST-IT note and put them on the wall.
- Step 2.** Group the similar customer requirements together.
- Step 3.** Review the customer requirements statements and throw out redundant ones.
- Step 4.** Write a summary to express the general idea for each group. For those that do not relate to anything else, label it as “independent.”
- Step 5.** Vote for the most important groups and rank the top three groups and assign some relationships. If a group supports another group in a positive manner, we add an arrow pointing from the supporting group to the supported group. If the relationship is contradictory, we add a line pointing between the two groups with blocks on the end.
- Step 6.** Look at each detailed customer requirement and highlight the new, unique, or difficult (NUD) ones.
- Step 7.** Ask customers to rank (on a scale of 1–10) the strength of importance for each requirement.

The result of these ranked and structured customer requirements will flow into the QFD process.

C. **Quality Function Deployment (QFD): The Houses of Quality (HOQ)** (see [14])

QFD is a methodology that establishes “bridges” between qualitative, high-level customer needs/requirements and the quantitative engineering terms that are critical to fulfilling these high-level needs. By following QFD, relationships can be explored among customer requirements, CTQ measures, Function Requirements (FRs), Design Parameters (DPs), and Process Variables (PVs), and the priorities of each CTQ, FR, DP, and PV can be quantitatively calculated.

Generally, the QFD methodology is deployed through a four-phase sequence.

- Phase 1 – critical-to-satisfaction planning (HOQ1)
- Phase 2 – functional requirements planning (HOQ2)
- Phase 3 – design parameters planning (HOQ3)
- Phase 4 – process variable planning (HOQ4)

In this chapter, HOQ 1 will be introduced in detail as an example.

Input:

Structured and ranked New, Unique, and Difficult (NUD) VOC from the KJ diagram

Key Output:

The priorities of each CTQ

How to do it (refer to [10, 15]):

- Step 1.** Convert NUD VOC (“WHATs”) into a list of CTQs (“HOWs”) in terms of the engineering perspective to support customer requirements along the roof of the house. There may be more than one CTQ to achieve each customer requirement.
- Step 2.** Quantify the relationship between each customer requirement to each CTQ on a 1-3-9 scale (9 = Strong fulfillment, 3 = Moderate fulfillment, 1 = Weak fulfillment, or 0 = No relationship). These values help to identify which CTQs are critical while some are not.
- Step 3.** Identify the correlation between each pair of CTQ to address the cooperative and conflicting relationships among CTQs to develop the design to be as cooperative as possible.
- Step 4.** Conduct a competitive assessment with a main competitor. The comparison with the key competitor on each customer requirement is on a 1–5 scale with 5 being high.
- Step 5.** Prioritize customer requirements. These priorities include importance to customer from the KJ method, improvement factor, and absolute weight. Customer requirements with low complete assessments and high importance are candidates for improvement, which will be assigned improvement factors on a 1–5 scale with 5 being the most essential target to improve. The absolute weight can then be calculated by multiplying the customer importance and the improvement factor.
- Step 6.** Priority CTQs. The CTQs are prioritized by determine absolute weight and relative weight. The absolute weight is calculated by the sum of the products of the relationship between customer requirements and CTQs and the importance to the customer. The relative weight is the sum of the products of the relationship between customer requirements and CTQs and customer requirements’ absolute weights. The relative and absolute weights are evaluated to prioritize and select CTQs for improvement.

Furthermore, the design team can apply the same method for identifying the relationship among CTQs, functional requirements, design parameters, and process variables.

D. **The Pugh Concept Evaluation and Selection Process** (See [16])

The Pugh concept evaluation is a solution-iterative selection process. The method alternates between generation and selection activities. The “generation” activity can be enriched by TRIZ (Theory of Inventive Problem Solving, see [17]) methodology to generate conceptual

solutions for each functional requirement. The “selection” activity can use a scoring matrix called the *Pugh Matrix* or the *criteria-based matrix* to evaluate the concepts.

Input:

Functional requirements and conceptual solutions to achieve corresponding FRs

Output:

The conceptual solutions, which are selected and ready to go forward into the design phase

How to do it (refer to [18, 19]):

- Step 1.** Define concept selection criteria from a clear and complete set of requirements.
- Step 2.** Define a best-in-class benchmarked datum concept.
- Step 3.** Provide candidate concepts to evaluate against the datum.
- Step 4.** Evaluate each concept against the datum using (+)s, (-)s, and (S)s to rank the fulfillment of the concept selection criteria.
(+) means the concept is better than the benchmarked datum concept.
(-) means the concept is worse than the benchmarked datum concept.
(S) means the concept is the same with the benchmarked datum concept.
- Step 5.** Refine criteria as necessary during the first cycle of evaluation.
- Step 6.** Analyze the results from the first cycle of evaluation: Sum of (+)s, (-)s, and (S)s.
- Step 7.** Identify weakness in concepts that can be turned into (+)s.
- Step 8.** Create hybrid “super concepts” by integrating the strengths of like concepts to remove (-) s and (S)s.
- Step 9.** Select a new datum based on the scoring that suggests a superior concept after the first cycle of evaluation.
- Step 10.** Add any new concepts that have been developed.
- Step 11.** Repeat the evaluation process through the second cycle.
- Step 12.** The superior concept is selected and ready to go forward into the development or design phase.

E. Design Failure Modes and Effects Analysis (DFMEA) (See [20])

DFMEA is applied to define qualitatively and rank quantitatively the failure modes and effects for new products and service processes across all the phases of DMADV. In particular, the design team can use DFMEA in a design concept for potential failure modes, so it can address them early in the design. Usually, DFMEA is

conducted on the superior concept, which is chosen from all the candidate concepts in the Pugh concept selection process.

Input:

Superior concept architectures, functional requirements, the physical form, etc.

Output:

Causes of failure and corrective action

How to do it (refer to [10, 19]):

- Step 1.** Develop a block diagram of the design element or function being analyzed (at system, subsystem, subassembly, or component level).
- Step 2.** Determine the ways in which each design element or function can fail (failure modes).
- Step 3.** Determine the effects of the failure on the customer(s) for each failure mode associated with the element or function.
- Step 4.** Identify potential causes of each failure mode.
- Step 5.** List the current design controls for each cause or failure mode.
- Step 6.** Assign severity, occurrence, and detection ratings to each cause.
- Step 7.** Calculate risk priority numbers (RPN) for each cause.
- Step 8.** Apply tools and best practices to reduce the sensitivity to root causes of failure or eliminate root causes of failure and recalculate RPNs.
- Step 9.** Document causes of failure and corrective action results qualitatively and quantitatively.

F. Response Surface Methods (See [7, 21])

Purpose:

Optimize the system performance in the *Design* phase by constructing a statistical model and response surface map that represents the relationship between the response and the critical design parameters. If the design parameters are quantitative and there are only a few of them, RSM is an effective tool for modeling and optimization.

Input:

Critical design parameters

Output:

Surface map and equations that determine the level of the factors

How to do it (refer to [22]):

- Step 1.** Choose a CTQ response to be studied by experimentation.

Step 2. Determine the critical parameter to be modified with the experiments. Focus on the significant factors that affect the response.

Step 3. Select the measurement system used to analyze the parameters.

Step 4. Create the transfer function from the experimental data. The transfer function is a mathematical description of the behavior of the system that can be used to create surface plots and optimize the system's performance.

Step 5. Plot the response surface maps to observe the system behavior.

Step 6. Final Output: a surface map and an equation that is used to determine the level of the factors. Sensitivity of the factors can also be analyzed.

G. Inferential Statistics

Inferential statistics is often employed in the *Verify* phase.

Purpose:

Identify and control variation in the critical responses.

Input:

The new product/service's performance data

Output:

The decision on which factors have an effect on the design's response

Definition:

Hypotheses and Risk: There are null hypothesis and alternate hypothesis. Once we have data, we can determine whether we should accept or reject the null hypothesis, by calculating a test statistic.

The t-Test: Used to compare two samples, or one sample with a standard. The null hypothesis is that the means are equal or the difference between the two population means is zero. For example, the t-test can be used to determine if two types of shoe sole materials wear differently, or whether the change to a process can affect the cycle time.

Analysis of Variance (ANOVA): The analysis of variance is a generalization of the two-sample t test which enables us to test for the significance of the difference among k ($k > 2$) factor and/or treatment means. A real scenario can be that a process engineer needs to assess the effect of three process times and three strength levels on the density of plastic. Another example is that an administrative team has four new form layouts to reduce data entry time on license agreements.

In the null hypothesis of ANOVA, the means of multiple populations or different treatments are assumed to be equal, whereas in the alternative hypoth-

esis, at least one mean is different. The key idea of ANOVA is that the total variation in the experiment data can be decomposed into two components: One is the between-treatment (factor) variation, and the other is the within-treatment (error) variation, i.e.,

$$SS_{\text{total}} = SS_{\text{treatment}} + SS_{\text{error}}$$

where SS refers to the sum of squares. If the between-treatment (factor) variation dominates, or the corresponding F score exceeds a threshold at a certain significance level, we will reject the null hypothesis and get a conclusion that the means of different treatments are significantly different.

H. Statistical Process Control (See [6, 9])

Purpose:

Monitor the critical response of the new product/service in the *Verify* phase to assess stability and predictability and detect important changes.

Input:

The new product/service's performance data

Output:

Assessment of the new product/service's stability, predictability, sigma level, and capability for commercialization readiness.

Main Considerations:

Sample size – sample size should be large enough to provide good sensitivity in detecting out-of-control conditions.

Sampling frequency – sampling should be frequent enough to ensure the opportunities for process control and improvement.

Concepts:

A production/service process that is operating with only chance causes (common causes) of variation present is said to be "in statistical control." A process is out of control if there exist assignable causes (special causes) that are not part of the chance cause pattern such as improperly adjusted or controlled machines, operator errors, or defective raw material [6]. An SPC chart is used to distinguish these two types of causes by upper and lower control limits (UCL and LCL). As long as all the sample points plot within the control limits, the process is assumed in statistical control. If a charting point is out of the control limits, this implies that there is evidence that the process is out of control. We then should investigate the assignable causes and take corrective actions.

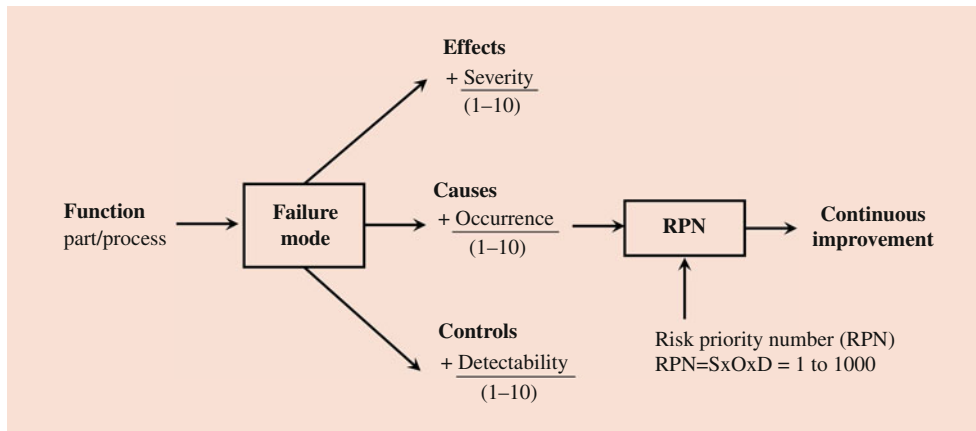


Fig. 13.2 Implementation flowchart of DFMEA

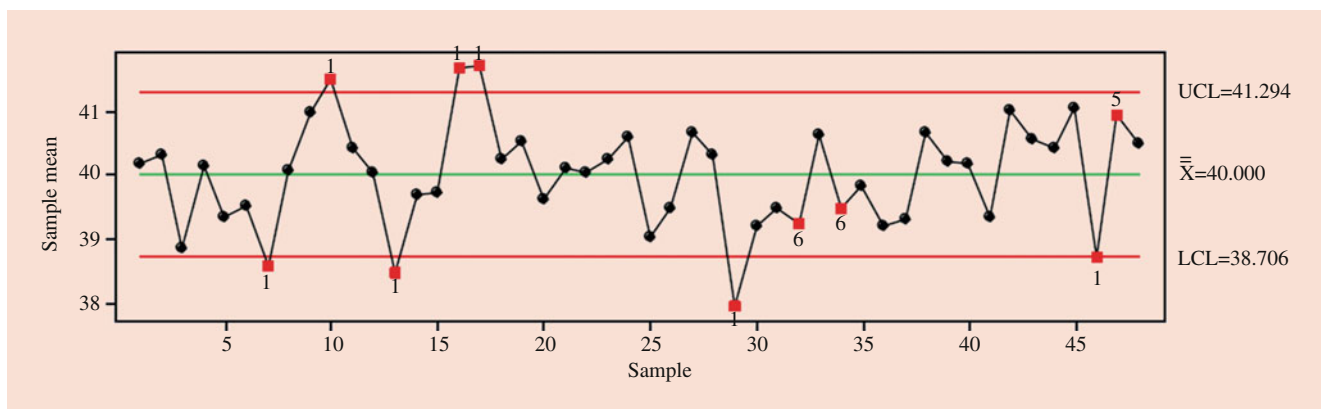


Fig. 13.3 An X-bar control chart

We can use SPC charts to determine if a new product/service's CTQ measure is in control. If it is, the product/service may move to the mass production phase.

How to do it (refer to [6]):

- Step 1.** Select the environment in which the data will be collected.
- Step 2.** Select the responses and parameters that will be monitored.
- Step 3.** Select the measurement systems used to acquire the data.
- Step 4.** Run the system in the prescribed environment and acquire the data.
- Step 5.** Plot the data using the appropriate type of SPC chart.
- Step 6.** Assess the plots for stability and predictability.
- Step 7.** Calculate the estimates of sigma level and process capability.

A typical X-bar control chart with sample size five for monitoring the mean of a process is given as an example

in Figs. 13.2 and 13.3, where the out-of-control points are highlighted as red squares.

13.4 Six Sigma Case Study

13.4.1 Introduction

In this section, a case study on printed circuit board (PCB) quality improvement by the authors is used to demonstrate the application of Six Sigma, which is digested from [2]. A more detailed report of this case may be found in [2]. This study was conducted in reference to the DMAIC approach, and the objective is to improve the sigma level for a series of product called PSSD in the screening process.

13.4.2 Process Background

This case study was conducted in an electronic company, which is located at an industrial park in southern China. The company manufactures multilayer PCB by using the surface

mount technology (SMT), which is a technique of placing surface mount devices (SMDs) on the surface of a PCB. SMDs are microminiature leaded or leadless electronic components that are soldered directly to the pads on the surface of the PCB. The major manufacturing processes in the company are solder screen, component placement, and solder reflow. As any defect in any of the solder joints can lead to the failure of the circuit, the screening process is regarded as the most critical process in the PCB manufacturing.

The screening process is a manufacturing process that transfers solder paste onto the solder pad of a PCB. The application method of solder paste is printing, and the printing technique used is off-contact printing in which there is a snap-off distance between a stencil and a PCB. The type of screening machine used to manufacture PSSD products is semiautomatic. During a printing process, two PCBs are placed side by side on the holder of a screening machine. The solder paste is then placed onto a stencil manually before printing. The front/back blade makes a line contact with the stencil and a close contact with the given amount of solder paste. The solder paste is then rolled in front of the front/back blade. In this way, solder paste is pressed against the stencil and transferred onto the solder pad through the stencil openings. More detailed operation of a screening process is described in [2].

13.4.3 Define Phase

In this case, we specifically focus on the improvement of the sigma level of the PCB screening process. In the screening process, the solder paste volume (height) transferred onto the PCB is the most important factor that needs to be controlled carefully. This is because too little solder paste can cause open circuits and too much solder paste can cause bridging between solder pads in the subsequent processes. As a result, the solder paste height on the solder pads is identified as a critical-to-quality (CTQ) characteristic (i.e., the y) that needs to be controlled in a very precise way by the company. According to that, a project team is formed and a project charter is constructed.

13.4.4 Measure Phase

To control the screening process, the project team in the company has asked operators to measure the solder paste height for the PSSD product on five different points on a PCB. The solder paste height on the five points is measured by using the Cyberoptics Cybersentry system every 4 hours. The gauge repeatability and reproducibility (R&R) of the measurement system was verified before the study on the solder paste height is conducted. The gauge R&R results

ensured that the data from the current measurement system are accurate enough for the following analysis.

13.4.5 Analyze Phase

Currently, six semiautomatic screening machines are used to manufacture the PSSD product. Therefore, the data on solder paste height of these six machines was collected from the company, and the process capability analysis was conducted for these screening machines in order to analyze the current printing performance. According to the analytical results, the process capability in machine no. 12 was not satisfactory because the capability index C_p was only 1.021, which was smaller than 1.33 (the four-sigma level). Moreover, another capability index C_{pk} was 0.387. This showed that the screening process was off-center. As shown in the capability plot in Fig. 13.4, we concluded that there exist both a high variance and a mean shift in the solder paste process. Therefore, we list all the potential factors (x 's) that may cause this through brainstorming and constructing a cause and effect diagram.

13.4.6 Improve Phase

In the analysis of the current printing performance, the result showed that the screening process capability of machine no. 12 was not satisfactory. After brainstorming with the mechanical engineers in the company, the designed experiments were decided to conduct on machine no. 12 in order to determine the optimal settings of all the input factors (x 's) in the screening process. In this phase, DOE was used as a core statistical tool for the sigma level improvement.

In the initial experiments, several possible factors that might have influence on the printing performance were taken into account. These experiments were used to screen out new factors that have influence on the printing performance. A full factorial experiment was carried out, and the considered

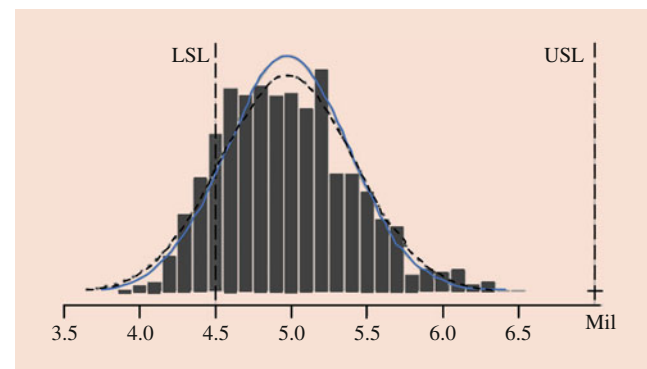
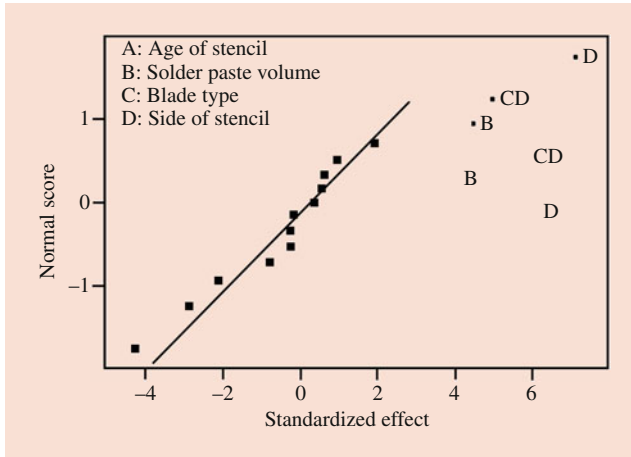


Fig. 13.4 Capability plot of machine no. 12

Table 13.2 Four factors and their levels in the initial experiments

Factor	Level 1	Level 2
Age of stencil	Old (25 months)	New (8 months)
Solder paste volume	Small	Large
Blade type	Front	Back
Side of stencil	Left	Right

**Fig. 13.5** Normal probability plot of the standardized effects**Table 13.3** The levels of each factor in the further experiments

Factor	Level 1	Level 2	Level 3
Solder paste viscosity	<150 mPa.s	>190 mPa.s	/
Speed of squeegee	0.4 inch/sec	0.7 inch/sec	1 inch/sec
Pressure of squeegee	18 bar	28 bar	/
Side of stencil	Left	Right	/
Blade type	Front	Back	/

four factors and their levels are given in Table 13.2. From the experimental results, the main effect of the solder paste volume and the side of stencil showed significant influence on the solder paste height. The interaction between the blade type and the side of stencil was also significant. These significant effects were supported by the normal probability plot of the standardized effects shown in Fig. 13.5.

These significant factors (blade type and side of stencil; solder paste volume was excluded since it was difficult to control the operator-to-operator variation in refilling the volume of the solder paste) would then be included together with the already-known significant factors (solder paste viscosity, speed of squeegee, and pressure of squeegee) in the further experiments. The aim of the further experiments was to determine the standard settings of all the significant factors (i.e., the vital few x 's). A full factorial experiment was carried out. The factors and their levels are given in Table 13.3.

To study the influence of these factors on the solder paste height and to draw conclusions, the main effect plots and the interaction plots were used. From the main effect plots shown in Fig. 13.6, the pressure of squeegee, blade type, and

side of stencil were significant factors for the height average. From the main effect plots shown in Fig. 13.7, the solder paste viscosity, speed of squeegee, blade type, and side of stencil were significant factors for the height variation. From the interaction plots for the height average shown in Fig. 13.8, the interaction between solder paste viscosity and blade type showed significant influence on the solder paste printing performance. The interaction between the speed of squeegee and blade type was also significant.

From the analytical results, the solder paste viscosity, speed of squeegee, blade type, and side of stencil were significant factors for the height variation. Therefore, low solder paste viscosity (<150 mPa.s), low speed of squeegee (0.4 inch/sec), front blade, and right side of stencil could result in small height variation. By using these optimal settings, the variation of the solder paste height can be reduced. After reducing the height variation, the sigma level of the screening process on machine no. 12 can be improved.

13.4.7 Control Phase

To sustain the improvement of the sigma level in the screening process, control plans for all the important x 's were proposed to the company. For example, both the CTQ y and the vital x 's should be monitored by SPC charts over time, so that the solder paste height variation and the sigma level can be controlled and sustained continuously. Also, the financial benefits through the reduction of COPQ were calculated.

The comparison of the printing performance before and after the project was reported in Table 13.4. After using the optimal settings, the sigma level of the screening process can be improved from 1.162 to 5.924. This shows that a nearly six-sigma performance can be achieved. According to [23], the level of defects per million opportunities (DPMO) would reduce to near 3.4 and the COPQ would be less than 1% of the sales. As a result, after the Six Sigma practice, the COPQ of the process for this company has been significantly reduced.

13.5 Six Sigma and Big Data

13.5.1 Background

The recent rapid advances in information technologies have reshaped today's business world into a digital regime. The widely deployed Internet-of-Things (IOT) infrastructure in both manufacturing and service industries generates massive amounts of data every moment, such as sensor data, images, videos, texts, etc. [24]. This coming Big Data era where data volume, variety, and velocity have all been increasing at an unprecedented scale is driving forward the next industrial revolution, also acknowledged as Industry 4.0. Many leading

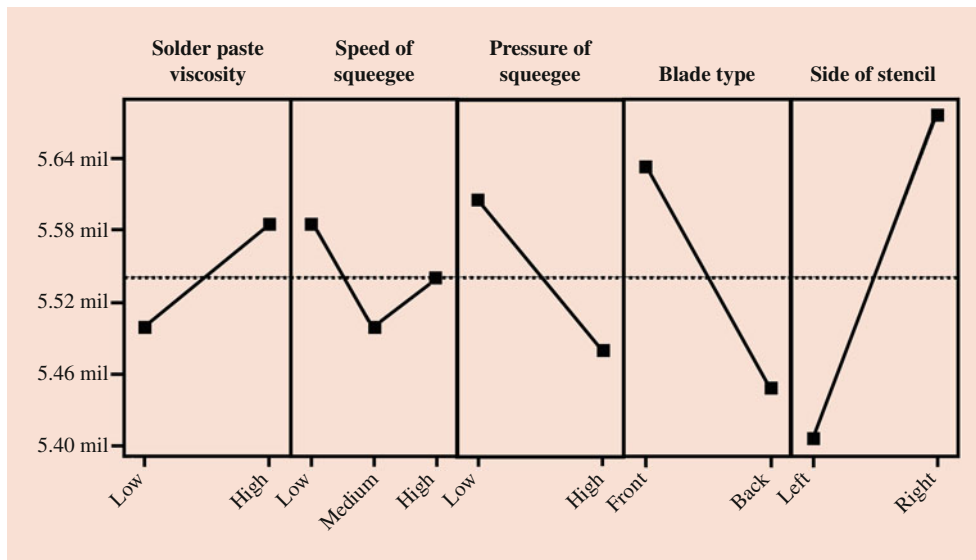


Fig. 13.6 Main effect plots for height average

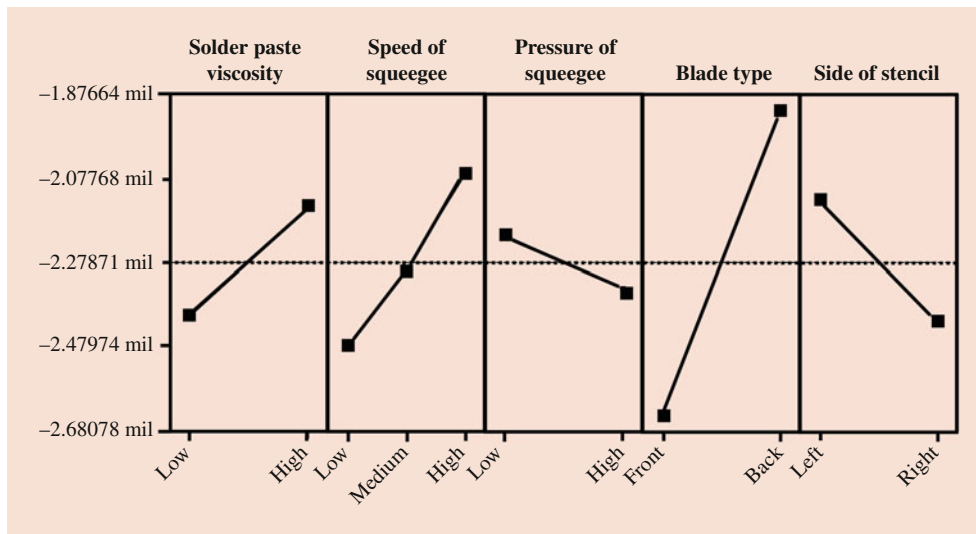


Fig. 13.7 Main effect plots for height variation

countries including the USA, Germany, China, and Japan have already launched their national strategies to catch this wave of data-enabled revolution.

It is believed that taking advantages of Big Data will become basic competition for today's enterprises. According to a report of McKinsey institute in 2011 [25], up to 50% decrease in product development and assembly costs will be achieved for global manufacturing sectors, and over 60% increase in net margin will be gained for retail markets. Big Data also provides an excellent opportunity for modern quality engineering and Six Sigma as data analytics has performed an essential role in their implementation. This created data-rich environment is pushing the traditional quality control and improvement practices toward an automatic,

intelligent, and integrated paradigm [26–32]. However, challenges also exist due to some intrinsic weaknesses of traditional quality control tools in Six Sigma:

- Most statistical analysis tools in Six Sigma, for example, inferential statistics and hypothesis testing, take sampling-based information which in fact only reflects partial status of underlying production or service processes.
- A universal quality data and information fusion center is not proposed and emphasized. Traditional existing databases in companies do not effectively integrate manufacturing information generated on the shop floor. Hence, it is usually very hard to share and track quality data.

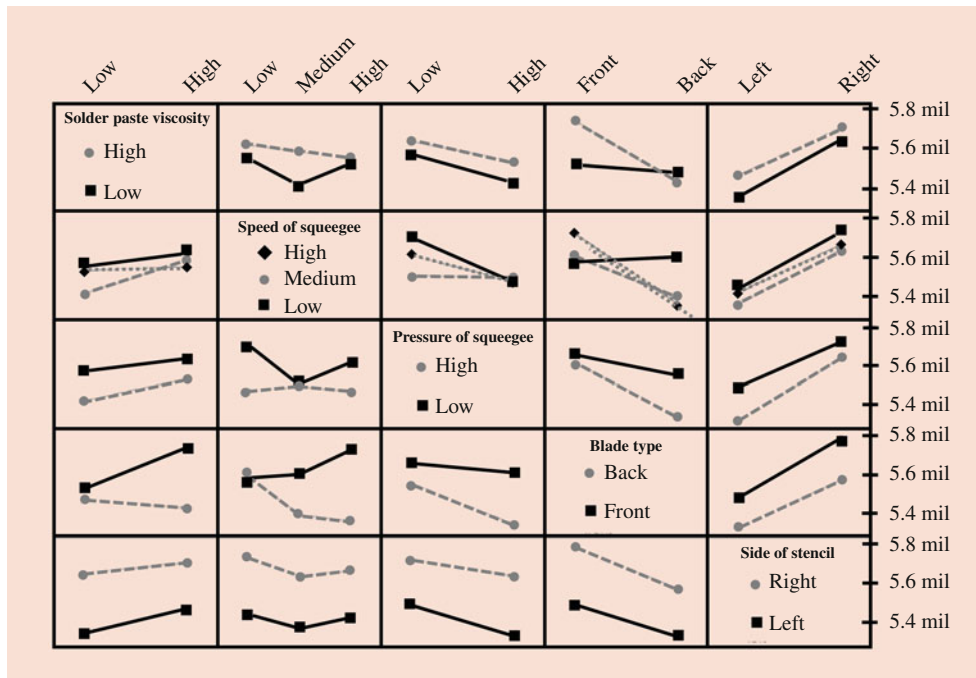


Fig. 13.8 Interaction plots for height average

Table 13.4 Comparison of printing performance before and after using optimal settings

	Mean	S.D.	Cp	Cpk	Sigma level	ppm	COQ
Before	4.974	0.408	1.021	0.387	1.162	122,173	>40% of sales
After	5.750	0.211	1.975	1.975	5.924	3.4	<1% of sales

- Typical Six Sigma tools such as FMEA and cause-effect diagrams for failure root causes analysis rely greatly on intensive human participation and subjective judgment. The automation and intelligence level of these tools are relatively low for efficient Big Data analytics.

Six Sigma has achieved extraordinary levels of success in productivity improvement and cost saving in almost every industry since the early 1990s. To accommodate the current Big Data era, the toolkit of Six Sigma has to be expanded and updated accordingly. *Journal of Quality Technology (JQT)*, a flagship refereed journal of the American Society for Quality (ASQ), has recently published three special issues in 2018 which discuss the potential applications of Big Data analytics in reliability and maintenance, statistical process control (SPC), and advanced manufacturing. We refer interested readers to the papers therein. In the following section, the new specific opportunities brought about by Big Data to the core methodologies of Six Sigma are outlined in detail.

13.5.2 DMAIC: New Opportunities

Six Sigma is usually implemented in a sequential and disciplined manner through five incremental phases, i.e., DMAIC, each phase of which is now faced with tremendous opportunities offered by Big Data to significantly enhance its capabilities. Generally speaking, data become new treasures and resources in an enterprise that should be leveraged by every quality participant from top leaders to floor operators. It is expected the combination of Big Data thinking and methodologies with Six Sigma will become popular in the very near future.

Phase 1. Define (D)

To define current Six Sigma projects, the capability of the Big Data infrastructure in the company should be taken into account in the project charter. The problem may be defined from a new perspective by using Big Data thinking, the objectives may be quantified more clearly, and more benefits can be expected with the usage of Big Data technologies. Additionally, specialists for information technology (IT engineers) as well as for data modeling and analysis (Data scientists) should both be incorporated into a Six Sigma team and collaborate with other engineers within that team. It has been observed by authors that, in many advanced industries, such as semiconductors and smart phones, data collection and knowledge discovery through the cooperation of IT engineers and data scientists has become a routine and essential part in quality initiatives.

Phase 2. Measure (M)

This phase will take more advantage of the available massive data to clarify the status of the processes of interest. The measure data nowadays have more complex characteristics:

- Quality data are no longer only univariate variables; they can be time series or profiles, product images, 3D scanned surfaces, surveillance videos, and customer comments. Each data type needs specific tools to extract relevant quality information.
- Quality data arrive at an extremely high speed, also termed as data streams. The high data velocity requires real-time responses in data collection, processing, and analysis.
- Data visualization, especially when data are of high dimension and huge size, becomes useful for practitioners to understand the inherent data patterns.

The effective handling of the above heterogeneous and massive data streams will generate a global and complete picture of the manufacturing and service processes, which also acts as a solid preparation for the Analyze phase in the following.

Phase 3. Analyze (A)

The Analyze phase will be enhanced to a large extent with the rapid development of machine learning tools. The data analytics has been evolving from the Descriptive level to Diagnostic and Predictive levels. Specifically, two main classes of machine learning tools can be adopted for Big Data analytics [33].

- Unsupervised learning. It discovers an internal representation or features based only on the input data or process variables (x 's). The common method is the clustering. The engineers can grasp the behavior and variation of the process variables to stabilize the processes.
- Supervised learning. It develops a relationship between process variables or input data (x 's) and product quality or output data (y). Upon this predictive model, product quality can be forecast beforehand and accidents can be early signaled. It includes a variety of powerful methods for either classification or regression purpose, such as support vector machines, neural networks, and ensemble methods.

The wide applications of the above machine learning tools, along with the emerging artificial intelligence (AI) methodologies, will definitely promote intelligence prosperity in Six Sigma.

Phase 4. Improve (I)

Quality improvement solutions in the Big Data era can be conducted in a real time basis. As the data are acquired and analyzed without time delay, the optimization solutions for the current process status can also be immediately given by using modern powerful parallel and cloud-computing technologies. The well-developed optimization theory, for example, the convex optimization and combinatorial optimization, can be introduced into this phase to seek for optimal improvement directions. Process variables are thus adjusted in a dynamic manner to maintain high quality level for every process. In a word, the traditional static improvement solution based on historical off-line data will be replaced by an adaptive alternative which flexibly responds to real-time online data.

Phase 5. Control (C)

To ensure the process improvement is sustainable, the Control phase monitors the new improved process to check if it is always in control. Big Data leads to high-dimension, large-scale, and real-time data streams that need to be effectively monitored. The SPC surveillance tools now have developed many computationally efficient versions for these challenges [34]. On the other hand, instead of waiting for abundant reference data before launching a monitoring scheme for a particular process, the collected data from similar processes can be borrowed to start the monitoring task as soon as possible by using current transfer learning techniques [35]. Once an out-of-control signal is issued, the fault diagnosis methods which combine the hypothesis testing, Bayesian network, and variable selection methods with Big Data can quickly track the process variables that have been changed to enable the root causes discovery and correction.

As a final note in this section, we summarize the maturity levels of Big Data analytics when it is applied to quality projects in Six Sigma. By referring to these levels, quality practitioners can have a better understanding of where they are at present and where they should head for in the future.

- Level 1. We only use experience, not data.
- Level 2. We collect data but just look at the numbers.
- Level 3. We group the data so as to form charts and graphs.
- Level 4. We use census data with descriptive statistics.
- Level 5. We use sample data with descriptive statistics.
- Level 6. We use sample data with inferential statistics.
- Level 7. We use real-time heterogeneous sensor data with descriptive statistics/visualization.
- Level 8. We use real-time heterogeneous sensor data with inferential/predictive statistics to support decision-making.

13.6 Conclusion

As Six Sigma is evolving over time, the advantages and benefits of other business excellence approaches can still be learned and utilized in future Six Sigma programs. According to [36], combining other tools or methodologies and the Six Sigma methodology may be a future trend. For example, combining Lean tools to the Six Sigma methodology has become popular during the last few years. And there are expected to be more combinations in the future.

These years, the Six Sigma efforts have been pushed to both the external suppliers and external customers along a supply chain, which have resulted in even larger overall business impacts and cost savings. We have also observed an increasing trend outside the USA, where more and more companies in Asia and Europe, including small-to-medium-sized enterprises, have been in various stages of Six Sigma deployment and discovered its far-reaching benefits. On the other hand, the coming Big Data era has shifted current quality projects in Six Sigma toward data-driven practices, and more new advanced tools in statistical learning, machine learning, and AI are needed to integrate Six Sigma and Big Data.

Acknowledgments The author thanks the HKUST Quality Lab student team for conducting an extensive review of Six Sigma for the input of this chapter. Tsung's research was supported by Hong Kong Research Grants Council (RGC) Grants 16201718 and 16203917. Wang's research was supported by Hong Kong Innovation and Technology Fund (ITF) Grant PiH/246/18.

References

1. Snee, R.D., Hoerl, R.W.: *Six Sigma Beyond the Factory Floor: Deployment Strategies for Financial Services, Health Care, and the Rest of the Real Economy*. Pearson Prentice Hall, New Jersey (2005)
2. Tong, J., Tsung, F., Yen, B.: A DMAIC approach for printed circuit board quality improvement. *Int. J. Adv. Manuf. Technol.* **23**, 523–531 (2004)
3. Ng, T.Y., Tsung, F., So, R.H.Y., Li, T.S., Lam, K.Y.: Six Sigma approach to reducing fall hazards among cargo handlers working on top of cargo containers: a case study. *Int. J. Six Sigma Competitive Advantage*. **1**(2), 188–209 (2005)
4. Pande, P.S., Neuman, R.P., Cavanagh, R.R.: *The Six Sigma Way: How GE, Motorola, and Other Top Companies Are Honing Their Performance*. McGraw-Hill, New York (2000)
5. Carlson, C.: *Effective FMEAs: Achieving Safe, Reliable, and Economical Products and Processes Using Failure Mode and Effects Analysis*. Wiley, New York (2012)
6. Montgomery, D.C.: *Statistical Quality Control: A Modern Introduction*, 7th edn. Wiley, New York (2009)
7. Wu, C.F.J., Hamada, M.: *Experiments: Planning, Analysis, and Parameter Design Optimization*, 2nd edn. Wiley, New York (2011)
8. Montgomery, D.C.: *Design and Analysis of Experiments*, 9th edn. Wiley, New York (2017)
9. Qiu, P.: *Introduction to Statistical Process Control*. Chapman and Hall/CRC, Boca Raton (2013)
10. El-haik, B., Roy, D.M.: *Service Design for Six Sigma: A Road Map for Excellence*. Wiley, New Jersey (2005)
11. Anton, J.: *Listening to the Voice of the Customer: 16 Steps to A Successful Customer Satisfaction Measurement Program*. Customer Service Group, New York (2005)
12. Kawakita, J.: *KJ Method: A Scientific Approach to Problem Solving*. Kawakita Research Institute, Tokyo (1975)
13. Spool, J.M.: The KJ-technique: A group process for establishing priorities. https://articles.uie.com/kj_technique/ (2004)
14. Cohen, L.: *Quality Function Deployment: How to Make QFD Work for You*. Addison Wesley, Boston (1995)
15. Maritan, D.: *Practical Manual of Quality Function Deployment*. Springer, Berlin, Heidelberg (2015)
16. Pugh, S.: *Total Design*. Addison Wesley, Boston, Massachusetts (1995)
17. Altshuller, G.S.: *Creativity as Exact Science*. Gordon & Breach, New York (1988)
18. Creveling, C.M., Slutsky, J.L., Antis, D.: *Design for Six Sigma in Technology and Product Development*. Prentice Hall PTR, New Jersey (2003)
19. Cudney, E.A., Agustiad, T.K.: *Design for Six Sigma: A Practical Approach Through Innovation*. CRC Press, Boca Raton, Florida (2016)
20. Stamatis, D.H.: *Failure Mode and Effect Analysis*. Quality Press, Seattle (1995)
21. Myers, R.H., Montgomery, D.C.: *Response Surface Methodology*, 4th edn. John Wiley & Sons, New York (2016)
22. Anderson, M.J., Whitcomb, P.J.: *RSM Simplified: Optimizing Processes Using Response Surface Methods for Design of Experiments*, 2nd edn. Productivity Inc., New York (2016)
23. Harry, M., Schroeder, R.: *Six Sigma*. Doubleday, New York (2000)
24. Yang, H., Kumara, S., Bukkapatnam, S.T., Tsung, F.: The internet of things for smart manufacturing: a review. *IISE Trans.* **51**(11), 1190–1216 (2019)
25. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Byers, A.H.: *Big Data: The Next Frontier for Innovation, Competition, and Productivity*. McKinsey Global Institute, Washington (2011)
26. Jiang, W., Wang, K., Tsung, F.: A variable-selection-based multivariate EWMA chart for process monitoring and diagnosis. *J. Qual. Technol.* **44**(3), 209–230 (2012)
27. Ning, X., Tsung, F.: Improved design of kernel distance-based charts using support vector methods. *IIE Trans.* **45**(4), 464–476 (2013)
28. Zhang, C., Tsung, F., Zou, C.: A general framework for monitoring complex processes with both in-control and out-of-control information. *Comput. Ind. Eng.* **85**, 157–168 (2015)
29. Ding, D., Tsung, F., Li, J.: Rank-based process control for mixed-type data. *IIE Trans.* **48**(7), 673–683 (2016)
30. Wang, A., Xian, X., Tsung, F., Liu, K.: A spatial-adaptive sampling procedure for online monitoring of big data streams. *J. Qual. Technol.* **50**(4), 329–343 (2018)
31. Wang, K., Tsung, F.: A fast and robust nonparametric monitoring scheme for free-form surface scanning data. *IEEE Trans. Autom. Sci. Eng.* **16**(4), 1675–1685 (2019)
32. Wang, K., Tsung, F.: Hierarchical sparse functional principal component analysis for multistage multivariate profile data. *IISE Trans.* **53**(1), 58–73 (2021)
33. Friedman, J., Hastie, T., Tibshirani, R.: *The Elements of Statistical Learning*. Springer, New York (2001)
34. Qiu, P.: Statistical process control charts as a tool for analyzing big data. In: Ahmed, S.E. (ed.) *Big and Complex Data Analysis*, pp. 123–138. Springer, New York (2017)

35. Tsung, F., Zhang, K., Cheng, L., Song, Z.: Statistical transfer learning: a review and some extensions to statistical process control. *Qual. Eng.* **30**(1), 115–128 (2018)
36. Hahn, G.J.: The future of six sigma. *ASQ Six Sigma Forum Mag.* **3**(3), 32–33 (2004)



Prof. Fugee Tsung received his PhD degree from the University of Michigan, Ann Arbor, MI, USA. He is currently a Chair Professor with the Department of Industrial Engineering and Decision Analytics in

Hong Kong University of Science and Technology, Hong Kong. His research interests include quality analytics in advanced manufacturing and service processes, industrial big data, and statistical process control.



Prof. Kai Wang received his PhD degree from Hong Kong University of Science and Technology (HKUST) in 2018. He is currently an Assistant Professor in the Department of Industrial Engineering, School of Management, Xi'an Jiaotong University, China. His research interests include statistical machine learning, transfer learning, profile data modeling and monitoring, and heterogeneous data fusion.



Statistical Models for Monitoring the High-Quality Processes

14

Min Xie , Thong Ngee Goh , and Tahir Mahmood

Contents

14.1	Introduction	262
14.2	Use of Exact Probability Limits	262
14.3	Control Charts Based on Cumulative Count of Conforming Items	263
14.3.1	CCC Type Control Charts.....	263
14.3.2	CCC- <i>r</i> Chart Based on Negative Binomial Distribution.....	264
14.4	Generalization of the <i>c</i>-Chart	265
14.4.1	Charts Based on the Zero-Inflated Poisson Distribution.....	266
14.4.2	Chart Based on the Generalized Poisson Distribution.....	268
14.5	Control Charts for the Monitoring of Time-Between-Events	269
14.5.1	CQC Chart Based on the Exponential Distribution	269
14.5.2	Chart Based on the Weibull Distribution.....	270
14.5.3	General <i>t</i> -Chart.....	271
14.6	Discussion	271
	References	272

Abstract

One important application of statistical models in the industry is *statistical process control*. Many control charts have been developed and used in the industry. They are easy to use but have been developed based on statistical principles. However, for today's high-quality processes, traditional control-charting techniques are not applicable in many situations. Research has been going on in the last few decades, and new methods have been proposed. This chapter summarizes some of these techniques.

High-quality processes are generally defined as those with very low defective rate or defect-occurrence rate, which is achieved in six sigma environment and in the advanced manufacturing environment. *Control charts* based on the cumulative count of conforming items are recommended for such processes. The use of such charts has opened up new frontiers in the research and applications of statistical control charts in general. In this chapter, several extended or modified statistical models are described. They are useful when the simple and basic geometric distribution is not appropriate or is insufficient.

In particular, we present some extended Poisson distribution models that can be used for count data with large numbers of zero counts. We also extend the chart to the case of general *time-between-events monitoring*; such an extension can be useful in service or reliability monitoring. Traditionally, the exponential distribution is used for the modeling of time-between-events, although other distributions such as the Weibull or gamma distribution can also be used in this context.

Keywords

CCC chart · CQC chart · Statistical process monitoring · Time-between-events · Zero-inflated models

M. Xie (✉)
Department of Systems Engineering and Engineering Management,
City University of Hong Kong, Hong Kong, China
e-mail: minxie@cityu.edu.hk

T. N. Goh
Department of Industrial Systems Engineering and Management,
National University of Singapore, Singapore, Republic of Singapore
e-mail: tng@nus.edu.sg

T. Mahmood
Industrial and Systems Engineering Department, College of
Computing and Mathematics, King Fahd University of Petroleum
and Minerals, Dhahran, Saudi Arabia
e-mail: tahir.mahmood@kfupm.edu.sa

14.1 Introduction

Control charting is one of the most widely used statistical techniques in the industry for process control and monitoring. It dates back to the 1920s when Walter Shewhart introduced the basic charting techniques in the United States [1]. Since then, it has been widely adopted worldwide, mainly in manufacturing and also in the service industries. The simplicity of the application procedure allows a non-specialist user to observe the data and plot the control chart for simple decision making. At the same time, it provides sophisticated statistical interpretation in terms of false-alarm probability and average run length, among other important statistical properties associated with decision-making based on sample information. The implementation of control charts had helped many companies to focus on important quality issues and problems such as those raised by out-of-control points on a control chart.

However, for high-quality or near-zero-defect processes, traditional *Shewhart charts* may not be suitable for process monitoring and decision-making. This is especially the case for Shewhart attribute charts [2]. Many problems, such as high false-alarm probability, inability to detect process improvement, unnecessary plotting of many zeros, etc., have been identified by various researchers [3–6]. To resolve these problems, new models and monitoring techniques have been developed recently.

Traditional charts are all based on the principle of the normal distribution, and the upper control limit (UCL) and lower control limit (LCL) are routinely computed as the mean plus and minus three times the standard deviation. That is, if the plotted quantity Y has mean μ and standard deviation σ , the control limits are given by

$$\text{UCL} = \mu + 3\sigma \quad \text{and} \quad \text{LCL} = \mu - 3\sigma. \quad (14.1)$$

Generally, when the distribution of Y is skewed, the probability of false alarm (FAR), i.e., the probability that a point indicating out-of-control when the process has actually not changed, is different from the nominal value of 0.0027 associated with a truly normal distribution. Note that for attribute charts, the plotted quantities usually follow a binomial or Poisson distribution, and this is far from the normal distribution unless the sample size is very large.

The purpose of this chapter is to review the important models and techniques that can be used to monitor high-quality processes. The procedure based on a general principle of the *cumulative count of conforming* items is first described; this is then extended to other distributions. The emphasis is on recent developments and also on practical methods that can be used by practitioners.

This chapter is organized as follows. First, the use of *probability limits* is described. Next, control charts based on monitoring of the cumulative count of conforming items

and simple extensions are discussed. Control charts based on the zero-inflated Poisson distribution and generalized Poisson distribution are then presented. These charts are widely discussed in the literature, and they are suitable for count or *attribute data*. For process monitoring, time-between-events monitoring is of growing importance, and we also provide a summary of methods that can be used to monitor process change based on time-between-events data. Typical models are the exponential, Weibull and gamma distribution.

14.2 Use of Exact Probability Limits

For high-quality processes, it is important to use *probability limits* instead of traditional *three-sigma limits*. This is true when the quality characteristic that is being plotted follows a skewed distribution. For any plotted quality characteristic Y , the probability limits LCL_Y and UCL_Y can be derived as

$$P(X < \text{LCL}_Y) = P(X > \text{UCL}_Y) = \alpha/2, \quad (14.2)$$

where α is the false-alarm probability, i.e., when the process is in-control (IC), the probability that the control chart raises the alarm. Assuming that the distribution $F(x)$ is known or has been estimated accurately from the data, the control limits can be computed.

Probability limits are very important for attribute charts as the quality characteristics are usually not normally distributed. If this is the case, the false-alarm probability could be much higher than the nominal value ($\alpha = 0.0027$ for traditional three-sigma limits). *Xie and Goh* [7] studied the exact probability limits calculated from the binomial distribution and the Poisson distribution applied for the *np chart* and the *c chart*.

For control-chart monitoring, the number of nonconforming items in samples of size n , assuming that the process fraction nonconforming is p , the probability that there are exactly k nonconforming items in the sample is

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}, \quad k = 0, 1, \dots, n \quad (14.3)$$

and the probability limits can be computed as

$$P(X \leq \text{LCL}) = \sum_{i=0}^{\text{LCL}} \binom{n}{i} p^i (1-p)^{n-i} = \frac{\alpha}{2} \quad (14.4)$$

and

$$P(X \leq \text{UCL}) = \sum_{i=0}^{\text{UCL}} \binom{n}{i} p^i (1-p)^{n-i} = 1 - \frac{\alpha}{2}. \quad (14.5)$$

As discussed, probability limits can be computed for any distribution and should be used when the distribution is

skewed. This will form the basis of the following discussion in this chapter. In some cases, although the solution is analytically intractable, they can be obtained with computer programs. It is advisable that probability limits be used unless the normality test indicates that deviation from a normal distribution is not significant.

14.3 Control Charts Based on Cumulative Count of Conforming Items

High-quality processes are usually characterized by low defective rates. In a near-zero-defect manufacturing environment, items are commonly produced and inspected one-by-one, sometimes automatically. We can record and use the cumulative count of conforming items produced before a nonconforming item is detected. This technique has been intensively studied in recent years.

14.3.1 CCC Type Control Charts

The idea of tracking the *cumulative count of conforming (CCC)* items to detect the onset of assignable causes in an automated (high-quality) manufacturing environment was first introduced by Calvin [13]. Goh [4] further developed this idea into what is known as the CCC charting technique. Some related discussions and further studies can be found in [8–14], among others. Xie et al. [15] provided extensive coverage of this charting technique and further analysis of this procedure.

CCC Chart Based on Geometric Distribution

For a process with a defective rate of p , the cumulative count of conforming items before the appearance of a nonconforming item say Y , follows a *geometric distribution*. This is given by

$$P(Y = n) = (1 - p)^{n-1}p, \quad n = 1, 2, \dots \quad (14.6)$$

The cumulative probability function of count Y is given by

$$P(Y \leq n) = \sum_{i=1}^n (1 - p)^{i-1}p = 1 - (1 - p)^n. \quad (14.7)$$

Assuming that the acceptable false-alarm probability is α , the probability limits for the CCC chart are obtained as

$$UCL = \ln(\alpha/2) / \ln(1 - p) \quad (14.8)$$

and

$$LCL = \ln(1 - \alpha/2) / \ln(1 - p) \quad (14.9)$$

Usually, the center line (CL) is computed as

$$CL = \ln(1/2) / \ln(1 - p). \quad (14.10)$$

Note that the decision rule is different from that of the traditional p or np chart. If a point is plotted above the UCL, the process is considered to have improved. When a point falls below the LCL, the process is judged to have deteriorated. An important advantage is that the CCC chart can detect not only the increase in the defective rate (process deterioration) but also the decrease in the defective rate (process improvement). Further, the relation between the CCC chart based on the geometric and binomial distribution was presented by Riaz et al. [16]. They showed that the survival function of the geometric distribution equals the probability of the binomial distribution. Hence, this relation leads to a new way for the monitoring of CCC charts.

The CUSUM Chart for the Cumulative Count of Conforming (CUSUM-CCC)

Chang and Gan [17] proposed a *Cumulative Sum (CUSUM) control chart*, which is capable of detecting the small number of shifts in the defective rate of the process. The upper and lower CUSUM statistics are defined as:

$$\begin{aligned} C_t^+ &= \max(0, k^+ - Y_t + C_{t-1}^+), \\ C_t^- &= \min(0, k^- - Y_t + C_{t-1}^-), \end{aligned} \quad (14.11)$$

where t indexes the order of observations and k^+ and k^- are the reference values which are based on the *sequential probability ratio test (SPRT)* approach as:

$$k = \frac{\ln[p_1(1 - p_0)/p_0(1 - p_1)]}{\ln[(1 - p_0)/(1 - p_1)]}. \quad (14.12)$$

where the p_0 is the IC defective rate while p_1 is the OOC defective rate. The decision limits h^+ and h^- for the respective charting statistics are obtained for the fixed value of α which is the inverse of the in-control average number of items sampled ($ANIS_0$). The process is declared as detroit when $C_t^+ < h^+$ and the process is declared as improved when $C_t^- < h^-$.

The EWMA Chart for the Cumulative Count of Conforming (EWMA-CCC)

The *Exponentially Weighted Moving Average (EWMA) control chart* is an alternative memory type chart, which is also effective than the Shewhart chart in detecting small changes in the population parameters. The EWMA control chart for the CCC items was proposed by Yeh et al., [18], and its statistic can be defined as:

$$E_t = wY_t + (1 - w)E_{t-1} \quad (14.13)$$

where, t indexes the order of observations and w is the weighted parameter of statistics which lies between zero and one (i.e., $0 < w \leq 1$). By using the properties of geometric series, the mean and variance of the EWMA statistics are defined as:

$$E(E_t) = \frac{1}{p_0}, \tag{14.14}$$

$$\text{Var}(E_t) = \frac{1}{p_0} \frac{w [1 - (1 - w)^{2t}]}{(2 - w)}. \tag{14.15}$$

Note that an increase in the defective rate may lead to a decrease in Y_t so, for the process deterioration, lower control limit (LCL_{E_t}) is defined as follows:

$$LCL_{E_t} = \frac{1}{p_0} - L_{E_t} \times \frac{1}{p_0} \sqrt{\frac{w [1 - (1 - w)^{2t}]}{(2 - w)}}, \tag{14.16}$$

where L_{E_t} is the control charting constant that depends on the choice of w and desired α . A Process is declared as detroit when $E_t < LCL_{E_t}$. Further, an optimal choice of this chart is discussed in [19].

A Simulated Example

For the application purpose, we have simulated 30 in-control (IC) data points with $p_0 = 0.0005$ and the last 20 out-of-control (OOC) data points with $p_1 = 0.00075$. The value of α is set to be 0.0000833 for the calculation of control limits (Table 14.1).

In Fig. 14.1, control charts for the CCC items are plotted, where Fig. 14.1a is about the typical CCC chart. For the conventional CCC chart, LCL is set at 0.08331423 and UCL is set at 20166.57. The amount of shift is too small that’s why CCC chart does not indicate any abnormality. Further, the CUSUM-CCC chart is plotted in Fig. 14.1b where $k^+ = 1622$ and $h^+ = 1607$ are set against the fixed p_0, p_1 , and α (See, Chang and Gan [17]). The findings of the CUSUM-CCC chart showed few FAR and detected all the OOC signals.

The EWMA-CCC chart is plotted in Fig. 14.1c where $w = 0.10$ and $L_{E_t} = 0.315$ are set against the fixed p_0 and α (See, Yeh et al. [18]). The results showed that EWMA-CCC suffered from few FAR as compared to CUSUM-CCC and all the OOC signals are detected.

Table 14.1 A set of cumulative count of conforming items data

3349	4020	1912	4201	339	543	2659	546	2027	820
2621	2199	1867	3597	1146	3097	2192	1009	5099	611
1950	586	1686	2472	942	3784	688	1753	1366	9
900	73	1984	41	1274	1002	9	34	1117	747
1348	1171	335	556	30	410	579	342	474	61

14.3.2 CCC-r Chart Based on Negative Binomial Distribution

A simple idea to generalize a CCC chart is to consider plotting of the cumulative count of items inspected until observing two nonconforming items. This was studied in [20] resulting in the CCC-2 control chart. This chart increases the sensitivity of the original CCC chart for the detection of small process shifts in p . The CCC-2 chart has smaller type II error, which is related to chart sensitivity, and steeper OC (Operating Characteristic) curves than the CCC chart with the same type I error, which is the false alarm probability. Further, Bersimis et al. [21] provided a compound methodology based on the CCC-1 and CCC-2 charts, which has better performance ability as compared to individual CCC-1 and CCC-2 charts.

A CCC-r chart [22, 23] plots the cumulative count of items inspected until r nonconforming items are observed. This will further improve the sensitivity and detect small changes faster. However, it requires more counts to be cumulated in order to generate an alarm signal. The CCC-r charting technique was also studied by Xie et al. [22].

Let Y be the cumulative count of items inspected until r nonconforming items have been observed. Suppose p the probability of an item to be nonconforming. Then Y follows a negative binomial distribution and its probability mass function is given by

$$P(Y = n) = \binom{n-1}{r-1} p^r (1-p)^{n-r}, \tag{14.17}$$

$n = r, r + 1, \dots$

The cumulative distribution function of count Y would be

$$F(n, r, p) = \sum_{i=r}^n P(Y = i) = \sum_{i=r}^n \binom{i-1}{r-1} p^r (1-p)^{i-r}. \tag{14.18}$$

If the acceptable false-alarm probability is α , then the upper control limit and the lower control limit, UCL_r and LCL_r , respectively, of the CCC-r chart can be obtained as the solution of the following equations:

$$F(UCL_r, r, p) = \sum_{i=r}^{UCL_r} \binom{i-1}{r-1} p^r (1-p)^{i-r} = 1 - \alpha/2 \tag{14.19}$$

and

$$F(LCL_r, r, p) = \sum_{i=r}^{LCL_r} \binom{i-1}{r-1} p^r (1-p)^{i-r} = \alpha/2. \tag{14.20}$$

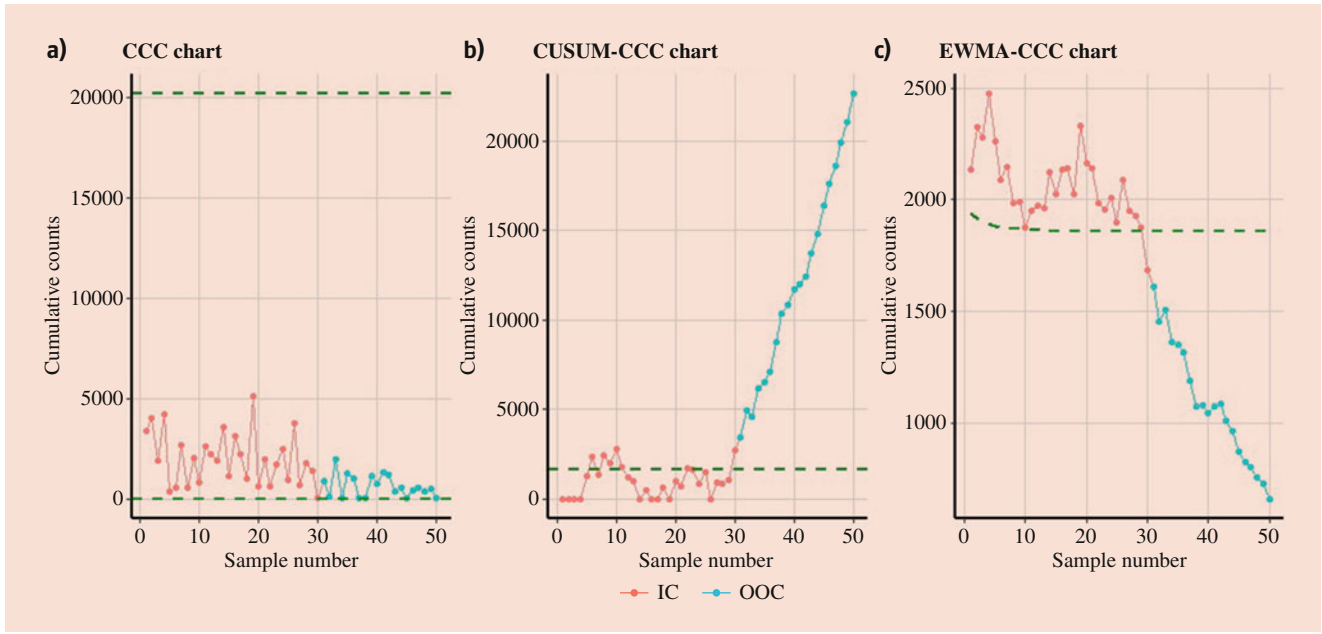


Fig. 14.1 Control charts based on CCC items

Note that this chart is suitable for the *one-by-one inspection process*, and so no subjective sample size is needed. On the other hand, the selection of r is a subjective issue if the cost involved is not a consideration. As the value of r increases, the sensitivity of the chart may increase, but the user probably needs to wait too long to plot a point. For this issue, *Albers [24]* designed a study to demonstrate the optimal value of r , which is related to the increase of p . *Ohta et al. [23]* addressed this issue from an economic design perspective and proposed a simplified design method to select a suitable value of r based on the *economic design method* for control charts that monitor discrete quality characteristics.

A $CCC-r$ chart provides better detection ability when a large amount of drifts are present in the process while to obtain the better detection ability on small to moderate amount of shifts, *Kotani et al. [25]* proposed $EWMA_{CCC-r}$ chart. The statistic of $EWMA_{CCC-r}$ chart can be defined as:

$$E_t = wY_t + (1 - w)E_{t-1} \quad (14.21)$$

where, Y_t is the cumulative count of items inspected until r nonconforming ones, t indexes the order of observations and w is the weighted parameters of statistics, which lies between zero and one (i.e., $0 < w \leq 1$). The mean and variance of the $EWMA_{CCC-r}$ statistics are defined as:

$$E(E_t) = \frac{r}{p}, \quad (14.22)$$

$$\text{Var}(E_t) = \frac{r(1-p)}{p^2} \frac{w[1 - (1-w)^{2t}]}{(2-w)}. \quad (14.23)$$

The decision lines of the $EWMA_{CCC-r}$ are defined as:

$$\begin{aligned} LCL_{E_t} &= \frac{r}{p} - L_{E_t} \times \frac{\sqrt{r(1-p)}}{p} \sqrt{\frac{w[1 - (1-w)^{2t}]}{(2-w)}}, \\ UCL_{E_t} &= \frac{r}{p} + L_{E_t} \times \frac{\sqrt{r(1-p)}}{p} \sqrt{\frac{w[1 - (1-w)^{2t}]}{(2-w)}}. \end{aligned} \quad (14.24)$$

where L_{E_t} is the control charting constant that depends on the choice of w and desired α . When E_t lies between LCL_{E_t} and UCL_{E_t} , the process is declared as stable process otherwise, process is said to be unstable. Further, the $EWMA_{CCC-r}$ is improved by *Kusukawa et al. [26]*.

14.4 Generalization of the c-Chart

The c-chart is based on monitoring of the number of defects in a sample. Traditionally, the number of defects in a sample follows the Poisson distribution. The control limits are computed as

$$UCL = c + 3\sqrt{c} \quad \text{and} \quad LCL = c - 3\sqrt{c}, \quad (14.25)$$

where c is the average number of defects in the sample, and the LCL is set to be zero when the value computed with (14.25) is negative.

However, for high-quality processes, it has been shown that these limits may not be appropriate. Some extensions of this chart are described in this section.

$$E(Y) = p\lambda \tag{14.28}$$

and

$$\text{Var}(Y) = p\lambda + p\lambda(\mu - p\lambda). \tag{14.29}$$

14.4.1 Charts Based on the Zero-Inflated Poisson Distribution

In a near-zero-defect manufacturing environment, many samples will have no defects. However, for those containing defects, we have observed that there could be many defects in a sample, and hence, the data has an over-dispersion pattern relative to the Poisson distribution. To overcome this problem, a generalization of Poisson distribution was used in [6, 27].

This distribution is commonly called the *zero-inflated Poisson (ZIP) distribution*. Let Y be the number of defects in a sample; the probability mass function is given by

$$\begin{cases} P(Y = 0) = (1 - p) + pe^{-\lambda} \\ P(Y = d) = p \frac{\lambda^d e^{-\lambda}}{d!} \quad d = 1, 2, \dots \end{cases} \tag{14.26}$$

This has an interesting interpretation. The process is basically *zero-defect*, although it is affected by causes that lead to one or more defects. If the occurrence of these causes is p , and the severity is λ , then the number of defects in the sample will follow a zero-inflated Poisson distribution.

The Shewhart Chart for ZIP Process

When the zero-inflated Poisson distribution provides an excellent fit to the data, two types of control charts can be applied. One is the exact probability limits control chart, and the other is the CCC chart. When implementing the exact probability limits chart, Xie and Goh [6] suggested that only the upper control limit n_u should be considered, since the process is in a near-zero-defect manufacturing environment and the probability of zero is usually very large. The upper control limit can be determined by:

$$\sum_{d=n_u}^{\infty} p \frac{\lambda^d e^{-\lambda}}{d!} \leq \alpha, \tag{14.27}$$

where α is the probability of the type I error. It should be noticed that n_u could easily be solved because it takes only discrete values.

Control charts based on the zero-inflated Poisson distribution commonly have better performance in the *near-zero-defect manufacturing* environment. However, the control procedure is more complicated than the traditional methods since more effort is required to test the suitability of this model with more parameters.

For the zero-inflation Poisson distribution, we have that [28]

It should be pointed out that the *zero-inflation Poisson model* is straightforward to use, as the mean and variance are of the closed-form. For example, the moment estimates can be obtained straightforward. On the other hand, the maximum-likelihood estimates can also be derived.

The maximum-likelihood estimates can be obtained by solving

$$\begin{cases} p = \frac{1 - n_0/n}{1 - \exp(-\lambda)} \\ \lambda = \bar{y}/p \end{cases} \tag{14.30}$$

where $\bar{y} = \sum_{i=1}^n y_i/n$ [28].

When the count data can be fitted by a zero-inflation Poisson model, statistical process control procedures can be modified. Usually, the lower control limit for zero-inflation Poisson model will not exist, because the probability of zero is larger than the predetermined type I error level. This is common for the *attribute control chart*. In the following section, the upper control limit will be studied.

The upper control limit n_u for a control chart based on the number of nonconformities can be obtained as the smallest integer solution of the following equation:

$$P(n_u \text{ or more nonconformities in a sample}) \leq \alpha_L, \tag{14.31}$$

where α_L is the predetermined false-alarm probability for the upper control limit n_u .

Here our focus is on data modeling with appropriate distribution. It can be noted that the model contains two parameters. To be able to monitor the change in each parameter, a single chart may not be appropriate. Xie and Goh [6] developed a procedure for the monitoring of individual parameter. First, a CCC chart is used for data with zero count. Second, a *c*-chart is used for those with one or more non-zero count. Note that a useful model should have practical interpretations. In this case, p is the occurrence probability of the problem in the process, and λ measures the severity of the problem when it occurs. Hence it is a useful model and important to be able to monitor each of these parameters so that any change from normal behavior can be identified.

Chang and Gan [9] extended the above-mentioned method by considering the *g* chart instead of the *c* chart and obtained better performance in terms of an *average number of samples (ANS)*. Furthermore, the effect of estimation errors in the ZIP control chart was discussed by [27]. Where

p considered as known in the whole study and the estimation effects of λ were discussed. It was recommended that at any given λ value, around 50–100 observations are enough to meet pre-determined FAR.

The MLE estimates of the ZIP distribution consists of the average \bar{y} which leads to misleading results when outliers exist in the dataset. Therefore, robust estimates of the ZIP distribution are necessary under extreme outliers. Hence, Yang et al. [29] and Li et al. [30] proposed control charts based on the robust estimators such as trimmed mean, winsorized mean and middle mean. Furthermore, Rakitzis and Castagliola [31] discussed the estimation effects of ZIP Shewhart control chart under Phase I study. They mentioned that there is no particular trend of historical samples regarding the estimation of appropriate ZIP parameters, but in most of the examined cases, it is necessary that the historical samples exceed 10^5 .

The CUSUM Chart for ZIP Process

The structure of a CUSUM chart consists of the previous and the current sample. Therefore, it is more sensitive to small and moderate parameter shifts. He et al. [32] proposed CUSUM structures for the ZIP distribution to tackle the simultaneous shifts in p and λ . Let p_0 and λ_0 are the IC parameters of the ZIP distribution than the p -CUSUM chart was designed to detect a shift from p_0 to p_1 . The p - CUSUM statistic is designed on the base of the likelihood ratio method, which is defined as follows:

$$B_i = \max(0, B_{i-1} + K_i); i = 1, 2, \dots, \tag{14.32}$$

where the initial value of the p - CUSUM statistic is set at zero and the K_i for the specific shift size is obtained as,

$$K_i = \begin{cases} \ln \frac{1-p_1+p_1e^{-\lambda_0}}{1-p_0+p_0e^{-\lambda_0}} & Y_i = 0 \\ \ln \frac{p_1}{p_0} & Y_i > 0 \end{cases} \tag{14.33}$$

The p - CUSUM chart resulted in a signal when B_i exceeds from the h_p . Where the h_p is selected to achieve the desired IC performance.

Further, the λ - CUSUM chart is designed to detect a shift from λ_0 to λ_1 and the statistic based on the likelihood ratio method is defined as follows:

$$L_i = \max(0, L_{i-1} + M_i); i = 1, 2, \dots \tag{14.34}$$

Where the initial value of the λ - CUSUM statistic is set at zero and the M_i for the specific shift size is obtained as,

$$M_i = \begin{cases} \ln \frac{1-p_1+p_1e^{-\lambda_1}}{1-p_0+p_0e^{-\lambda_0}} & Y_i = 0 \\ Y_i \ln \frac{\lambda_1}{\lambda_0} + (\lambda_0 - \lambda_1) & Y_i > 0 \end{cases} \tag{14.35}$$

The λ - CUSUM chart signals when L_i exceeds from the h_λ . Where the h_λ is also selected to achieve the desired IC performance. The p - λ CUSUM chart is further obtained by the combination of p - CUSUM and λ - CUSUM charts.

For the simultaneous monitoring of the ZIP parameters p and λ . They also proposed another chart known by the t - CUSUM chart. The t - CUSUM chart is designed to detect shifts from p_0 to p_1 and λ_0 to λ_1 . The statistic for t - CUSUM chart is also based on the likelihood ratio method, which is defined as follows:

$$T_i = \max(0, T_{i-1} + N_i); i = 1, 2, \dots \tag{14.36}$$

where the initial value of the t - CUSUM statistic is set at zero and the N_i for the specific shift size is obtained as,

$$N_i = \begin{cases} \ln \frac{1-p_1+p_1e^{-\lambda_1}}{1-p_0+p_0e^{-\lambda_0}} & Y_i = 0 \\ Y_i \ln \frac{\lambda_1}{\lambda_0} + (\lambda_0 - \lambda_1) + \ln \frac{p_1}{p_0} & Y_i > 0 \end{cases} \tag{14.37}$$

The t - CUSUM chart signals when T_i exceeds from the h_t . Where the h_t is also selected to achieve the desired IC performance. Their findings revealed that the t - CUSUM structure outperforms all other charts in all increasing shifts except the increase in parameter p while the p - λ CUSUM structure is effective to detect inverse shifts in both parameters (i.e., one increase while another decrease). Although the t - CUSUM structure is useful to discuss the simultaneous shifts but using this structure, it is very difficult to diagnose and interpret OOC signals.

Furthermore, He et al. [33] proposed a combined CUSUM structure to monitor a ZIP process. In this combined method, a CUSUM chart was designed for the zero part of the process based on Bernoulli distribution while other CUSUM chart was developed for the non-zero portion of the process by using zero-truncated Poisson distribution. The comparison of the combined method is made with simultaneous charts proposed by [32], and findings depicted that the proposed method outperformed all other methods under discussion.

The EWMA Chart for ZIP Process

Fatahi et al. [34] proposed an EWMA structure for the monitoring of ZIP parameters. The proposed structure was designed on the moment method estimates which are defined as,

$$\hat{\lambda} = \frac{\sum_{t=1}^m (Y_t^2 - Y_t)}{\sum_{t=1}^m Y_t}, \tag{14.38}$$

and

$$\hat{p} = \frac{(\sum_{t=1}^m Y_t^2)^2}{m \sum_{t=1}^m (Y_t^2 - Y_t)}. \tag{14.39}$$

Where m denotes the number of subgroups in the Phase I samples. The statistic of ZIP-EWMA chart can be defined as:

$$E_t = wY_t + (1 - w) E_{t-1} \tag{14.40}$$

where, t indexes the order of observations and w is the weighted parameter of statistics which lies between zero and one (i.e., $0 < w \leq 1$). The mean and variance of the ZIP-EWMA statistics are defined as:

$$E(E_t) = \hat{\lambda}\hat{p}, \tag{14.41}$$

$$\text{Var}(E_t) = \left[\hat{\lambda}\hat{p} + \hat{\lambda}\hat{p} (\hat{\lambda} - \hat{\lambda}\hat{p}) \right] \frac{w [1 - (1 - w)^{2t}]}{(2 - w)}. \tag{14.42}$$

The decision lines of the ZIP-EWMA chart are defined as:

$$\text{LCL}_{E_t} = \hat{\lambda}\hat{p} - L_{E_t} \times \left[\hat{\lambda}\hat{p} + \hat{\lambda}\hat{p} (\hat{\lambda} - \hat{\lambda}\hat{p}) \right] \sqrt{\frac{w [1 - (1 - w)^{2t}]}{(2 - w)}},$$

$$\text{UCL}_{E_t} = \hat{\lambda}\hat{p} + L_{E_t} \times \left[\hat{\lambda}\hat{p} + \hat{\lambda}\hat{p} (\hat{\lambda} - \hat{\lambda}\hat{p}) \right] \sqrt{\frac{w [1 - (1 - w)^{2t}]}{(2 - w)}}. \tag{14.43}$$

where L_{E_t} is used to decide the width of control limits and depends on the choice of w and desired α . When E_t lies between LCL_{E_t} and UCL_{E_t} , the process is declared as stable process, otherwise the process is said to be unstable. Further, this method is extended by [35] into a combined structure and a simultaneous structure of EWMA chart (alternative method of simultaneous CUSUM chart proposed by [33]) for the monitoring of ZIP parameters.

An Example

An example is used here for illustration [2]. The data set used in Table 14.2 is the read-write errors discovered in a computer hard disk in the manufacturing process.

For the data set in Table 14.2, it can be seen that it contains many samples with no nonconformities. From the data set, the maximum-likelihood estimates are $\hat{p} = 0.1346$ and $\hat{\mu} = 8.6413$. The overall zero-inflation Poisson model for the data set is

$$f(y) = \begin{cases} 1 - 0.1346 + 0.1346 \exp(-8.6413), & \text{if } y = 0, \\ 0.1346 \frac{8.6413^y \exp(-8.6413)}{y!}, & \text{if } y > 0. \end{cases} \tag{14.44}$$

Table 14.2 A set of defect count data

0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	6	0	9
11	0	1	2	0	0	0	0	0	0	0	0	0	0	3	3	0	0	5	0	15	6
0	0	0	4	2	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
75	0	0	0	0	75	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0
0	2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0
0	0	1	0	0	0	0	0														

For the data set in Table 14.2, it can be calculated that the upper control limit is 14 at an acceptable false-alarm rate of 0.01. This means that there should not be an alarm for values less than or equal to 14 when the underlying distribution model is a zero-inflated Poisson distribution.

Note that for the CUSUM structures of ZIP distribution, real-life example about LED packaging process is discussed in He et al. [32, 33] while the ZIP-EWMA chart with health-related illustration is presented in Fatahi et al. [34].

14.4.2 Chart Based on the Generalized Poisson Distribution

The generalized Poisson distribution is another useful model that extends the traditional Poisson distribution, which only has one parameter. A two-parameter model is usually much more flexible and able to model different types of data sets. Since in the situation of over-dispersion or under-dispersion, the Poisson distribution is no longer preferable as it must have equal mean and variance, the generalized Poisson distribution [36] can be used.

This distribution has two parameters (θ, λ) and the probability mass function is defined as

$$P_X(\theta, \lambda) = \frac{\theta(\theta + x\lambda)^{x-1} e^{-\theta-x\lambda}}{x!}, \quad x = 0, 1, 2, \dots, \tag{14.45}$$

where $\lambda, \theta > 0$.

For the generalized Poisson distribution, we have that [36]

$$E(X) = \theta(1 - \lambda)^{-1} \tag{14.46}$$

and

$$\text{Var}(X) = \theta(1 - \lambda)^{-3}. \tag{14.47}$$

It should be pointed out that the generalized Poisson distribution model is straightforward to use as both the mean and variance are of closed-form. For example, the moment estimates can easily be calculated. On the other hand, the maximum-likelihood estimates can also be obtained straightforwardly. Consider a set of observation (X_1, X_2, \dots, X_n) with sample size n , the maximum-likelihood estimation $(\hat{\theta}, \hat{\lambda})$ can be obtained by solving

$$\begin{cases} \sum_{i=1}^n \frac{x_i(x_i - 1)}{\bar{x} + (x_i - \bar{x})\hat{\lambda}} - n\bar{x} = 0, \\ \hat{\theta} = \bar{x}(1 - \hat{\lambda}). \end{cases} \quad (14.48)$$

Here a similar approach as for the zero-inflated Poisson model can be used. One could also develop two charts for practical monitoring. One chart can be used to monitor the severity and another to monitor the dispersion or variability in terms of the occurrence of defects.

Furthermore, Chen et al. [37] proposed three control charts named by the *Shewhart GZIP chart*, ranked probability chart (RPC), and GZIP-CUSUM chart. The findings of the study revealed that the shape of the distribution is severely affected by λ and behave almost similar to a change in parameter θ . When relative difference among λ 's was small, then similar performance was observed for the both Shewhart GZIP and RPC charts, while in the presence of large differences, the RPC performed well as compared to all other charts. The GZIP-CUSUM chart performed well when the shifts are introduced in θ , while for the shifts in λ , the *GZIP-CUSUM chart* has satisfactory performance. In general, with probability control limits, some values are omitted that may have relatively high occurrence possibility, and some are included that may have a relatively small occurrence possibility. In such a case, probability control limits have a credibility issue which is improved by [38] using the narrowest confidence interval for the Shewhart GZIP chart.

An Example

The data in Table 14.2 can also be modeled with a *generalized Poisson distribution*. Based on the data, the maximum-likelihood estimates can be computed as $\hat{\theta} = 0.144297$ and $\hat{\lambda} = 0.875977$. The overall generalized Poisson distribution model for the data set is

$$f(x) = \frac{0.144297(0.144297+0.875977x)^{x-1}}{x!} \times \frac{e^{-0.144297-0.875977x}}{x!}, \quad x = 0, 1, 2, \dots \quad (14.49)$$

With this model, it can be calculated that the upper control limit is 26 at a false-alarm rate of 0.01. This means that there should not be any alarm for the values less than or equal to 26 when the underlying distribution model is the generalized

Poisson distribution. It should be mentioned here that, for this data set, both models can fit the data well, and the traditional Poisson distribution is rejected by statistical tests.

Recently, *Mahmood and Xie* [39] designed a comprehensive review of the control charts based on the *zero-inflated models* and their applications. Moreover, they have provided some future topics which are not covered in the monitoring methodologies of zero-inflated models.

14.5 Control Charts for the Monitoring of Time-Between-Events

Chan et al. [40] proposed a charting method called the cumulative quantity control chart (*CQC chart*). Suppose that defects in a process are observed according to a Poisson process with a mean rate of occurrence equal to $\lambda (>0)$. Then the number of units Q required to observe exactly one defect is an exponential random variable. The control chart for Q can be constructed to monitor possible shifts of λ in the process, which is the CQC chart.

The CQC chart has several advantages. It can be used for low-defective-rate processes as well as moderate-defective-rate processes. When the process defect rate is low or moderate, the CQC chart does not have the shortcoming of showing up frequent false alarms. Furthermore, the CQC chart does not require rational grouping of samples. The data required is the time between defects or defective items. This type of data is commonly available in equipment and process monitoring for production and maintenance.

When process failures can be described by a *Poisson process*, the time between failures will be exponential, and the same procedure can be used in reliability monitoring. Here we briefly describe the procedure for this type of monitoring. Since time is our preliminary concern, the control chart will be termed a *t-chart* in this chapter. This is in line with the traditional *c-chart* or *u-chart*, to which our *t-chart* may be a more suitable alternative. In fact, the notation also makes it easier for the extension to be discussed later.

14.5.1 CQC Chart Based on the Exponential Distribution

The distribution function of the with parameter λ is given by

$$F(t; \lambda) = 1 - e^{-\lambda t}, \quad t \geq 0. \quad (14.50)$$

The control limits for *t-chart* are defined in such a manner that the process is considered to be out of control (OOC) when the time to observe exactly one failure is less than the lower control limit (LCL), T_L , or greater than the upper control limit (UCL), T_U . When the behavior of the process

is normal, there is a chance for this to happen, and it is commonly known as a false alarm. The traditional false-alarm probability is set to be 0.27%, although any other false-alarm probability can be used. The actual acceptable false-alarm probability should, in fact, depend on the actual product or process. Assuming an acceptable probability for false alarms of α , the control limits can be obtained from the exponential distribution as:

$$T_L = \lambda^{-1} \ln \frac{1}{1 - \alpha/2} \quad (14.51)$$

and

$$T_U = \lambda^{-1} \ln \frac{2}{\alpha}. \quad (14.52)$$

The median of the distribution is the center line (CL), T_C , and it can be computed as

$$T_C = \lambda^{-1} \ln 2 = 0.693\lambda^{-1}. \quad (14.53)$$

These control limits can then be utilized to monitor the *failure times* of components. After each failure, the time can be plotted on the chart. If the plotted point falls between the calculated control limits, this indicates that the process is in the state of statistical control and no action is warranted. If the point falls above the upper control limit, this indicates that the process average, or the failure occurrence rate, may have decreased, resulting in an increase in the time between failures. This is an important indication of possible process improvement. If this happens, the management should look for possible causes for this improvement, and if the causes are discovered, then action should be taken to maintain them. If the plotted point falls below the lower control limit, this indicates that the process average, or the failure occurrence rate, may have increased, resulting in a decrease in the failure time. This means that the process may have deteriorated and thus, actions should be taken to identify and remove them.

In either case, the people involved can know when the reliability of the system has changed, and by a proper follow-up, they can maintain and improve the reliability. Another advantage of using the control chart is that it informs the maintenance crew when to leave the process alone, thus saving time and resources.

The CQC chart based on *exponential distribution* under CUSUM structure was studied by Gan [41] and Gan [42] extended this chart under EWMA structure to obtained better performance on the small or moderate shifts in the process. However, computation of run-length properties of exponential EWMA chart was discussed by Gan and Chang [43], and a comparative study based on these charts was reviewed by Liu et al. [43, 44]. Further, the robustness of this chart was discussed in [45, 46], and the estimation effects were

discussed in [47, 48]. Recently, [49, 50] extended this chart into other competitive charting methodologies.

14.5.2 Chart Based on the Weibull Distribution

It is well known that the lifetime distribution of many components follows a *Weibull distribution* [51]. Hence when monitoring reliability or equipment failure, this distribution has been shown to be very useful. The Weibull distribution function is given as

$$F(t) = 1 - \exp \left[- \left(\frac{t}{\theta} \right)^\beta \right], \quad t \geq 0, \quad (14.54)$$

where $\theta > 0$ and $\beta > 0$ are the so-called scale parameter and shape parameter, respectively.

The Weibull distribution is a generalization of the exponential distribution, which is recovered for $\beta = 1$. Although the exponential distribution has been widely used for *time-between-events*, Weibull distribution is more suitable as it is more flexible and is able to deal with different types of the *aging phenomenon in reliability*. Hence in reliability monitoring of equipment failures, the Weibull distribution is a good alternative.

A process can be monitored with a control chart, and the time-between-events can be used. For the Weibull distribution, the control limits can be calculated as:

$$UCL = \theta_0 \left[\ln \left(\frac{2}{\alpha} \right) \right]^{1/\beta_0} \quad (14.55)$$

and

$$LCL = \theta_0 \left[\ln \left(\frac{2}{2 - \alpha} \right) \right]^{1/\beta_0}, \quad (14.56)$$

where α is the acceptable false-alarm probability, and β_0 and θ_0 are the in-control shape and scale parameter, respectively. Generally, the false-alarm probability is fixed at $\alpha = 0.0027$, which is equivalent to the *three-sigma limits* for an X-bar chart under the normal-distribution assumption.

The center line can be defined as

$$CL = \theta_0 [\ln 2]^{1/\beta_0}, \quad (14.57)$$

Xie et al. [52] carried out some detailed analysis of this procedure. Since this model has two parameters, a single chart may not be able to identify changes in a parameter. However, since in a reliability context, it is unlikely that the shape parameter will change, and it is the scale parameter that could be affected by ageing or wear, a control chart as shown in Fig. 14.2 can be useful in reliability monitoring.

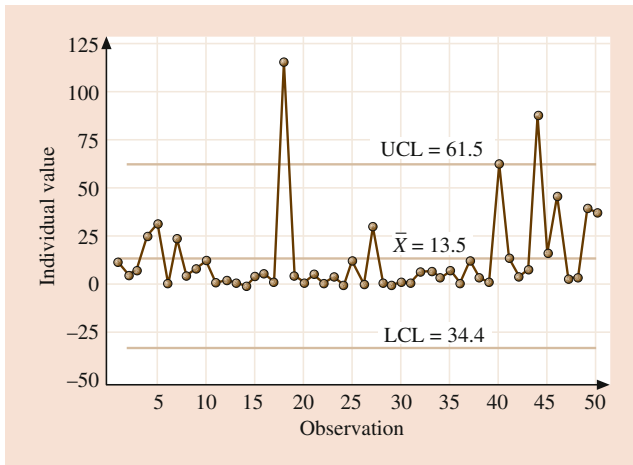


Fig. 14.2 A set of Weibull data and the plot

The *CUSUM* structure for the study variable, which follows the Weibull distribution was designed by Shafae et al. [53], and the *mixed EWMA-CUSUM* chart was developed by Aslam [54]. Further, a comparative study on the charts based on Weibull distribution was discussed by Wang et al. [55], and advanced methods related to this chart were presented in [56–58] and references therein.

14.5.3 General t-Chart

In general, to model *time-between-events*, any distribution for positive random variables could be used. Which distribution is used should depend on the actual data, with the exponential, Weibull and Gamma being the most common distributions. However, these distributions are usually very skewed. The best approach is to use probability limits. It is also possible to use a transformation so that the data is transformed to near-normality so that traditional chart for individual data can be used; such charting procedure is commonly available in *statistical process control* software.

In general, if the variable Y follows the distribution $F(t)$, the probability limits can be computed as usual, that is:

$$F(LCL_Y) = 1 - F(UCL_Y) = \alpha/2, \quad (14.58)$$

where α is the fixed false-alarm rate. This is an approach that summarizes the specific cases described earlier. However, it is important to be able to identify the distribution to be used.

Furthermore, to make better use of the traditional monitoring approach, we could use a simple normality transformation. The most common ones are the *Box-Cox transformation* and the log or power transformations. They can be easily realized in software such as MINITAB. Figure 14.2 shows a chart for a Weibull-distributed process characteristic and Fig. 14.3 shows the individual chart with a Box-Cox transformation.

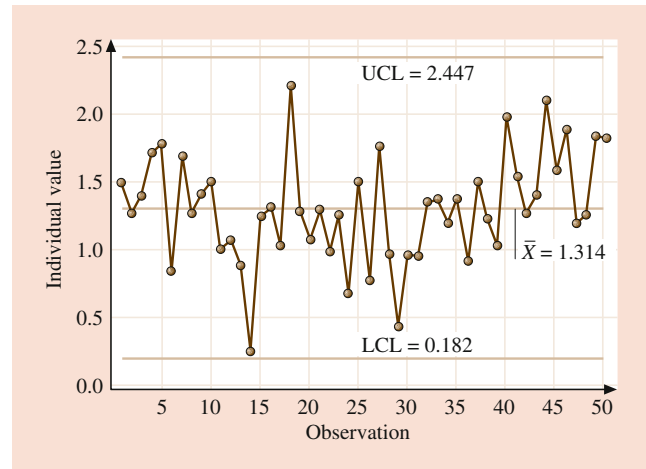


Fig. 14.3 The same data set as in Fig. 14.2 with the plot of the Box-Cox transformation

Recently, real time monitoring of the *time-between-events* applied to several bio-medical applications [59] and discussion on these methods are made in [60, 61].

The traditional t chart is used to monitor the time-between-events while X chart is used to monitor the magnitude of the events. Recently, few researchers suggested *monitoring magnitude* and time-between-events together. Similar to the \bar{X} & R or \bar{X} & S chart, Wu et al., [62] provided a combined scheme T & X chart for the *joint monitoring* of magnitude and time-between-events. Further, the rate chart was proposed by Wu et al. [63] which is more efficient method than the T & X chart while a comparative study for combined structure was designed by Rahali et al. [64]. These proposals were designed by assuming the independence between magnitude and time-between-events which may not hold in real situations. Hence, to alleviate this problem Cheng et al., [65] provided a simultaneous monitoring method based on *bivariate gamma distribution*. Furthermore, joint monitoring methods based on the distance and max-type measures were proposed by Sanusi and Mukherjee [66] which is a flexible scheme for any parametric distribution of *magnitude and time-between-events*.

14.6 Discussion

In this chapter, some effective control-charting techniques are described. The statistical monitoring technique should be tailored to the specific distribution of the data that are collected from the process. Perfunctory use of the traditional chart will not help much in today's manufacturing environment toward the near-zero-defect process. For high-quality processes, it is more appropriate to monitor items inspected between two nonconforming items or the time between two events.

The focus of this chapter is to highlight some common statistical distributions for process monitoring. Several

statistical models such as the geometric, negative binomial, zero-inflated Poisson, and generalized Poisson can be used for count-data monitoring in this context. The exponential, Weibull and Gamma distributions can be used to monitor time-between-events data, which is common in reliability or equipment failure monitoring. Other general distributions of time-between-events can also be used when appropriate. The approach is still simple: by computing the probability limits for a fixed false-alarm probability, any distribution can be used in a similar way. The simple procedure is summarized below:

- Step 1. Study the process and identify the statistical distribution for the process characteristic.
- Step 2. Collect data and estimate the parameters (and validate the model, if needed).
- Step 3. Compute the probability limits or use an appropriate normality transformation with an individual chart.
- Step 4. Identify any assignable cause and take appropriate action.

The distributions presented in this chapter open the door to further implementation of statistical process control techniques in a near-zero-defect era. Several research issues remain. For example, the problem with correlated data and the estimation problem has to be studied. In a high-quality environment, *failure or defect data* is rare, and the estimation problem becomes severe. In the case of continuous production and measurement, data correlation also becomes an important issue. It is possible to extend the approach to consider the exponentially weighted moving average (EWMA) or cumulative-sum (CUSUM) charts that are widely advocated by statisticians. A further area of importance is multivariate quality characteristics. However, a good balance between statistical performance and ease of implementation and understanding by practitioners is essential.

References

1. Shewhart, W.A.: Economic Control of Quality of Manufactured Product. ASQ Quality Press (1931)
2. Xie, M., Goh, T.N.: Some procedures for decision making in controlling high yield processes. *Qual. Reliab. Eng. Int.* **8**(4), 355–360 (1992)
3. Calvin, T.: Quality control techniques for “zero defects”. *IEEE Trans. Compon. Hybrids Manuf. Technol.* **6**(3), 323–328 (1983)
4. Goh, T.N.: A charting technique for control of low-defective production. *Int. J. Qual. Reliab. Manage.* **4**(1), 53–62 (1987)
5. Goh, T.N.: Statistical monitoring and control of a low defect process. *Qual. Reliab. Eng. Int.* **7**(6), 479–483 (1991)
6. Xie, M., Goh, T.N.: Improvement detection by control charts for high yield processes. *Int. J. Qual. Reliab. Manage.* **10**(7) (1993)
7. Xie, M., Goh, T.N.: The use of probability limits for process control based on geometric distribution. *Int. J. Qual. Reliab. Manage.* **14**(1), 64–73 (1997)
8. Bourke, P.D.: Detecting a shift in fraction nonconforming using run-length control charts with 100% inspection. *J. Qual. Technol.* **23**(3), 225–238 (1991)
9. Chang, T.C., Gan, F.F.: Charting techniques for monitoring a random shock process. *Qual. Reliab. Eng. Int.* **15**(4), 295–301 (1999)
10. Glushkovsky, E.A.: On-line g-control chart for attribute data. *Qual. Reliab. Eng. Int.* **10**(3), 217–227 (1994)
11. Kaminsky, F.C., Bennayan, J.C., Davis, R.D., Burke, R.J.: Statistical control charts based on a geometric distribution. *J. Qual. Technol.* **24**(2), 63–69 (1992)
12. Quesenberry, C.P.: Geometric Q charts for high quality processes. *J. Qual. Technol.* **27**(4), 304–315 (1995)
13. Wu, Z., Yeo, S.H., Fan, H.T.: A comparative study of the CRL-type control charts. *Qual. Reliab. Eng. Int.* **16**(4), 269–279 (2000)
14. Xie, W., Xie, M., Goh, T.N.: Control charts for processes subject to random shocks. *Qual. Reliab. Eng. Int.* **11**(5), 355–360 (1995)
15. Xie, M., Goh, T.N., Ranjan, P.: Some effective control chart procedures for reliability monitoring. *Reliab. Eng. Syst. Saf.* **77**(2), 143–150 (2002)
16. Riaz, M., Abbas, N., Mahmood, T.: A communicative property with its industrial applications. *Qual. Reliab. Eng. Int.* **33**(8), 2761–2763 (2017)
17. Chang, T.C., Gan, F.F.: Cumulative sum charts for high yield processes. *Stat. Sin.* **11**(3), 791–806 (2001)
18. Yeh, A.B., Mcgrath, R.N., Sembower, M.A., Shen, Q.: EWMA control charts for monitoring high-yield processes based on non-transformed observations. *Int. J. Prod. Res.* **46**(20), 5679–5699 (2008)
19. Mavroudis, E., Nicolas, F.: EWMA control charts for monitoring high yield processes. *Commun. Stat. Theory Methods.* **42**(20), 3639–3654 (2013)
20. Chan, L.Y., Xie, M., Goh, T.N.: Two-stage control charts for high yield processes. *Int. J. Reliab. Qual. Saf. Eng.* **4**(2), 149–165 (1997)
21. Bersimis, S., Koutras, M.V., Maravelakis, P.E.: A compound control chart for monitoring and controlling high quality processes. *Eur. J. Oper. Res.* **233**(3), 595–603 (2014)
22. Xie, M., Lu, X.S., Goh, T.N., Chan, L.Y.: A quality monitoring and decision-making scheme for automated production processes. *Int. J. Qual. Reliab. Manage.* **16**(2), 148–157 (1999)
23. Ohta, H., Kusakawa, E., Rahim, A.: A CCC-r chart for high-yield processes. *Qual. Reliab. Eng. Int.* **17**(6), 439–446 (2001)
24. Albers, W.: The optimal choice of negative binomial charts for monitoring high-quality processes. *J. Statist. Plann. Inference.* **140**(1), 214–225 (2010)
25. Kotani, T., Kusakawa, E., Ohta, H.: Exponentially weighted moving average chart for high-yield processes. *Ind. Eng. Manag. Syst.* **4**(1), 75–81 (2005)
26. Kusakawa, E., Kotani, T., Ohta, H.: A synthetic exponentially weighted moving average chart for high-yield processes. *Ind. Eng. Manag. Syst.* **7**(2), 101–112 (2008)
27. He, B., Xie, M., Goh, T.N., Ranjan, P.: On the estimation error in zero-inflated Poisson model for process control. *Int. J. Reliab. Qual. Saf. Eng.* **10**(02), 159–169 (2003)
28. Böhning, D.: Zero-inflated Poisson models and CA MAN: a tutorial collection of evidence. *Biom. J.* **40**(7), 833–843 (1998)
29. Yang, J., Xie, M., Goh, T.N.: Outlier identification and robust parameter estimation in a zero-inflated Poisson model. *J. Appl. Stat.* **38**(2), 421–430 (2011)
30. Li, D.Y., Yang, J., Li, M., Zhang, X.: Control chart based on middle mean for fine manufacturing process. *Adv. Mater. Res.* **339**, 406–410 (2011)
31. Rakitzis, A.C., Castagliola, P.: The effect of parameter estimation on the performance of one-sided Shewhart control charts for zero-inflated processes. *Commun. Stat. Theory Methods.* **45**(14), 4194–4214 (2016)

32. He, S., Huang, W., Woodall, W.H.: CUSUM charts for monitoring a zero-inflated poisson process. *Qual. Reliab. Eng. Int.* **28**(2), 181–192 (2012)
33. He, S., Li, S., He, Z.: A combination of CUSUM charts for monitoring a zero-inflated Poisson process. *Commun. Stat. Simul. Comput.* **43**(10), 2482–2497 (2014)
34. Fatahi, A.A., Noorossana, R., Dokouhaki, P., Moghaddam, B.F.: Zero inflated Poisson EWMA control chart for monitoring rare health-related events. *J Mech Med Biol.* **12**(04), 1250065 (2012).
35. Leong, R.N.F., Tan, D.S.Y.: Some zero inflated Poisson-based combined exponentially weighted moving average control charts for disease surveillance. *The Philipp. Stat.* **64**(2), 17–28 (2015)
36. Consul, P.C.: *Generalized Poisson Distributions: Properties and Applications*. Marcel Dekker, New York (1989)
37. Chen, N., Zhou, S., Chang, T.S., Huang, H.: Attribute control charts using generalized zero-inflated Poisson distribution. *Qual. Reliab. Eng. Int.* **24**(7), 793–806 (2008)
38. Yang, J., Xie, M., Goh, T.N.: Control limits based on the narrowest confidence interval. *Commun. Stat. Theory Methods.* **40**(12), 2172–2181 (2011)
39. Mahmood, T., Xie, M.: Models and monitoring of zero-inflated processes: the past and current trends. *Qual. Reliab. Eng. Int.* **35**(8), 2540–2557 (2019).
40. Chan, L.Y., Xie, M., Goh, T.N.: Cumulative quantity control charts for monitoring production processes. *Int. J. Prod. Res.* **38**(2), 397–408 (2000)
41. Gan, F.: Design of optimal exponential CUSUM control charts. *J. Qual. Technol.* **26**(2), 109–124 (1994)
42. Gan, F.: Designs of one-and two-sided exponential EWMA charts. *J. Qual. Technol.* **30**(1), 55–69 (1998)
43. Gan, F., Chang, T.: Computing average run lengths of exponential EWMA charts. *J. Qual. Technol.* **32**(2), 183–187 (2000)
44. Liu, J., Xie, M., Goh, T., Sharma, P.: A comparative study of exponential time between events charts. *Qual. Technol. Quant. Manag.* **3**(3), 347–359 (2006)
45. Borrór, C.M., Keats, J.B., Montgomery, D.C.: Robustness of the time between events CUSUM. *Int. J. Prod. Res.* **41**(15), 3435–3444 (2003)
46. Pehlivan, C., Testik, M.C.: Impact of model misspecification on the exponential EWMA charts: a robustness study when the time-between-events are not exponential. *Qual. Reliab. Eng. Int.* **26**(2), 177–190 (2010)
47. Ozsán, G., Testik, M.C., Weiß, C.H.: Properties of the exponential EWMA chart with parameter estimation. *Qual. Reliab. Eng. Int.* **26**(6), 555–569 (2010)
48. Zhang, M., Megahed, F.M., Woodall, W.H.: Exponential CUSUM charts with estimated control limits. *Qual. Reliab. Eng. Int.* **30**(2), 275–286 (2014)
49. Yen, F.Y., Chong, K.M.B., Ha, L.M.: Synthetic-type control charts for time-between-events monitoring. *PLoS One.* **8**(6), e65440 (2013).
50. Ali, S.: Time-between-events control charts for an exponentiated class of distributions of the renewal process. *Qual. Reliab. Eng. Int.* **33**(8), 2625–2651 (2017)
51. Murthy, D.N.P., Xie, M., Jiang, R.: *Weibull Models*. Wiley, New York (2003)
52. Xie, M., Goh, T.N., Kuralmani, V.: *Statistical Models and Control Charts for High Quality Processes*. Kluwer Academic, Boston (2002)
53. Shafae, M.S., Dickinson, R.M., Woodall, W.H., Camelio, J.A.: Cumulative sum control charts for monitoring Weibull-distributed time between events. *Qual. Reliab. Eng. Int.* **31**(5), 839–849 (2015)
54. Aslam, M.: A mixed EWMA–CUSUM control chart for Weibull-distributed quality characteristics. *Qual. Reliab. Eng. Int.* **32**(8), 2987–2994 (2016)
55. Wang, F.K., Bizuneh, B., Abebe, T.H.: A comparison study of control charts for Weibull distributed time between events. *Qual. Reliab. Eng. Int.* **33**(8), 2747–2759 (2017)
56. Chong, K.M.B., Xie, M.: A study of time-between-events control chart for the monitoring of regularly maintained systems. *Qual. Reliab. Eng. Int.* **25**(7), 805–819 (2009)
57. Aslam, M., Arif, O.H., Jun, C.-H.: An attribute control chart for a Weibull distribution under accelerated hybrid censoring. *PLoS One.* **12**(3), e0173406 (2017).
58. Aslam, M., Azam, M., Jun, C.-H.: A HEWMA-CUSUM control chart for the Weibull distribution. *Commun. Stat. Theory Methods.* **47**(24), 5973–5985 (2018)
59. Sparks, R., Jin, B., Karimi, S., Paris, C., MacIntyre, C.: Real-time monitoring of events applied to syndromic surveillance. *Qual. Eng.* **31**(1), 73–90 (2019)
60. Megahed, F.M.: Discussion on “Real-time monitoring of events applied to syndromic surveillance”. *Qual. Eng.* **31**(1), 97–104 (2019)
61. Wilson, J.D.: Discussion on “Real-time monitoring of events applied to syndromic surveillance”. *Qual. Eng.* **31**(1), 91–96 (2019)
62. Wu, Z., Jiao, J., He, Z.: A control scheme for monitoring the frequency and magnitude of an event. *Int. J. Prod. Res.* **47**(11), 2887–2902 (2009)
63. Wu, Z., Jiao, J., He, Z.: A single control chart for monitoring the frequency and magnitude of an event. *Int. J. Prod. Econ.* **119**(1), 24–33 (2009)
64. Rahali, D., Castagliola, P., Taleb, H., Khoo, M.B.: Evaluation of Shewhart time-between-events-and-amplitude control charts for several distributions. *Qual. Eng.* **31**(2), 240–254 (2019)
65. Cheng, Y., Mukherjee, A., Xie, M.: Simultaneously monitoring frequency and magnitude of events based on bivariate gamma distribution. *J. Stat. Comput. Simul.* **87**(9), 1723–1741 (2017)
66. Sanusi, R.A., Mukherjee, A.: A combination of max-type and distance based schemes for simultaneous monitoring of time between events and event magnitudes. *Qual. Reliab. Eng. Int.* **35**(1), 368–384 (2019)



Min Xie received his PhD in Quality Technology from Linköping University, Sweden in 1987. Currently he is Chair Professor of Industrial Engineering at City University Hong Kong and was a Professor at National University Singapore. Prof Xie has authored several book and many journal papers, and he was elected fellow of IEEE in 2005 and elected ISI member in 2010.



Thong Ngee Goh (Ph.D. University of Wisconsin – Madison, USA) is recipient of ASQ, Statistics Division’s William G Hunter Award (2007) and Eugene L Grant Medal (2012). He was Director, Office of Quality Management, NUS, Singapore. Specializing in statistical methodologies for engineering applications, he serves on the editorial boards of several leading research journals on Quality Management and Quality Engineering.



Tahir Mahmood got his degree of BS (Hons.) in Statistics with distinction (Gold Medalist) from the Department of Statistics, University of Sargodha, Sargodha, Pakistan. In 2015, he secured a scholarship from Deanship of Graduate Studies, King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia. In April 2017, he received his MS (Applied Statistics) degree from Department of Mathematics and Statistics, KFUPM. He received his Ph.D. (Systems Engineering) from the Department of Systems Engineering and Engineering Management, City University of Hong Kong, Hong Kong. In the past, he has served as a Lecturer in the Department of Technology, School of Science and Technology, The Hong Kong Metropolitan University, Hong Kong. Nowadays, he is an Assistant Professor in the Industrial and Systems Engineering Department, College of Computing and Mathematics, King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia. His current research interests include Statistical Process Monitoring, Quality Management, and Quality Engineering.



Statistical Management and Modeling for Demand Spare Parts

15

Emilio Ferrari , Arrigo Pareschi, Alberto Regattieri ,
and Alessandro Persona

Contents

15.1	The Forecast Problem of Spare Parts	276
15.1.1	Exponential Smoothing.....	277
15.1.2	Croston's Method.....	277
15.1.3	Holt Winter Models.....	277
15.2	Forecasting Methods	279
15.2.1	Characterizing Forecasting Methods.....	279
15.3	Forecasting Methods Applicability Related to Spare Parts Demands	280
15.4	Prediction of Aircraft Spare Parts: A Case Study ...	280
15.5	Poisson Models	284
15.5.1	Stock Level Conditioned to Minimal Availability Method.....	284
15.5.2	Stock Level Conditioned to Minimum Total Cost.....	285
15.6	Models Based on Binomial Distribution	286
15.6.1	An Industrial Application.....	287
15.7	Binomial Model Extension Based on Total Cost Function	289
15.7.1	Service Level Optimization: Minimum Total Cost Method.....	289
15.7.2	Simulation and Results.....	290
15.7.3	An Industrial Application.....	290
15.8	Weibull Extension	292
15.8.1	The Modified Model Extension by Weibull Distribution.....	292
15.8.2	Simulation and Results.....	293
15.9	A Differentiated Approach for Spare Parts: Standard Versus Custom-Made Components	295
15.9.1	Performance Analysis of Statistical Models.....	296
15.9.2	Performance Analysis of Poisson and Binomial Models.....	299

15.9.3	Operational Management of Spare Parts.....	301
--------	--------------------------------------------	-----

References	302
-------------------------	-----

Abstract

In recent years, increased emphasis has been placed on improving decision-making in business and government. A key aspect of decision-making is being able to predict the circumstances surrounding individual decision situations. Examining the diversity of requirements in planning and decision-making situations, it is clearly stated that no single forecasting methods or narrow set of methods can meet the needs of all decision-making situations. Moreover, these methods are strongly dependent on factors, such as data quantity, pattern, and accuracy, that reflect their inherent capabilities and adaptability, such as intuitive appeal, simplicity, ease application, and, least but not last, cost.

Section 15.1 deals with the placement of demand forecasting problem as one of biggest challenge in the repair and overhaul industry; after this brief introduction, Sect. 15.2 summarizes the most important categories of forecasting methods; paragraphs from 15.3 to 15.4 approach the forecast of spare parts first as a theoretical construct, but some industrial applications and results are added from field training, as in many other parts of this chapter.

Section 15.5 undertakes the question of optimal stock level for spare parts, with particular regards to Low Turnaround Index (LTI) parts conceived and designed for the satisfaction of a specific customer request, by the application of classical Poisson methods of minimal availability and minimum cost; similar considerations are drawn and compared in Sect. 15.6 dealing with models based on binomial distribution. An innovative extension of binomial models based on total cost function is discussed in Sect. 15.7. Finally, Sect. 15.8 adds the Weibull failure rate function to LTI spare parts stock level in maintenance system with declared wear conditions.

E. Ferrari (✉) · A. Pareschi · A. Regattieri
Department of Industrial Engineering, University of Bologna,
Bologna, Italy
e-mail: emilio.ferrari@unibo.it; arrigo.pareschi@unibo.it;
alberto.regattieri@unibo.it

A. Persona
Department of Management and Engineering, University of Padova,
Vicenza, Italy

Keywords

Spare parts management · Forecasting · Demand plan · Materials requirement · Optimal stock level

15.1 The Forecast Problem of Spare Parts

Demand forecasting is one of the most crucial issues of inventory management. Forecasts, which form the basis for the planning of inventory levels, are probably the biggest challenge in the repair and overhaul industry.

An example is in the airlines industry, when a common problem is the need to forecast short-term demand with the highest possible degree of accuracy. The high cost of modern aircraft and the expense of such repairable spares, as aircraft engines and avionics, greatly contributes to the considerable total investment of many airline operators. These parts, though low in demand, are critical to operations, and their unavailability can lead to excessive down time costs.

This problem is absolutely relevant in case of intermittent demand. A demand of an item is classified as *intermittent* when irregular and sporadic. This type of demand, typical for a great amount of spare parts, is very difficult to predict. This complicates the efficient management and control of the inventory system, which requires an acceptable balance between inventory costs, on the one hand, and stock-outs, on the other. Inventory management models require accurate forecasts in order to true this balance up.

We can consider explicitly both the demand pattern and the size of the demand when it occurs for the classification of the data demand patterns and classify them into four categories [1], as follows:

- *Intermittent demand*, which appears randomly with many time periods having no demand.
- *Erratic demand*, which is (highly) variable, erratic relating to the demand size rather than demand per unit time period.
- *Slow moving* (“smooth”) demand, which also occurs at random with many time periods having no demand. Demand, when it occurs, is for single or very few units.
- *Lumpy demand*, which likewise seems random with many time periods having no demand. Moreover, demand, when it occurs, is (highly) variable. Lumpy concept corresponds to an extremely irregular demand, with great differences between each period’s requirements and with a great number of periods with zero requirements.

Traditionally, the characteristics of intermittent demand are derived from two parameters: the average inter-demand interval (ADI) and the coefficient of variation (CV). ADI measure the average number of time periods between two

successive demands, and CV represents the standard deviation of period requirements divided by the average period requirements.

$$CV = \frac{\sqrt{\frac{\sum_{i=1}^n (\varepsilon_{ri} - \varepsilon_a)^2}{n}}}{\varepsilon_a} \tag{15.1}$$

where n is the number of periods and ε_{ri} and ε_a are real and average demand of spare parts in period i . The four resulting demand categories are graphically presented in Fig. 15.1.

The categorization scheme should suggest the different ways to treat the resulting categories according to the following characteristics:

- The “ $ADI \leq x; CV^2 \leq y$ ” condition tries effectively to test for stock-keeping units, SKUs, which are not very intermittent and erratic (i.e., faster moving parts or parts whose demand pattern does not raise any significant forecasting or inventory control difficulties).
- The “ $ADI > x; CV^2 \leq y$ ” condition tests for low-demand patterns with constant, or more generally, no highly variable demand sizes (i.e., not very erratic).
- The “ $ADI > x; CV^2 > y$ ” condition tests for lumpy demand items. Lumpy demand may be defined as a demand with great differences between each period’s requirements and with a great number of periods having zero requests.
- Finally, the “ $ADI \leq x; CV^2 > y$ ” condition tests for erratic (irregular) demand items with rather frequent demand occurrences (i.e., not very intermittent).

In all these cases, x denotes the ADI cutoff value ($ADI = 1.32$) measuring the average number of time periods between two successive demands and y the corresponding CV^2 cutoff value ($CV^2 = 0.49$) that is equal to the square

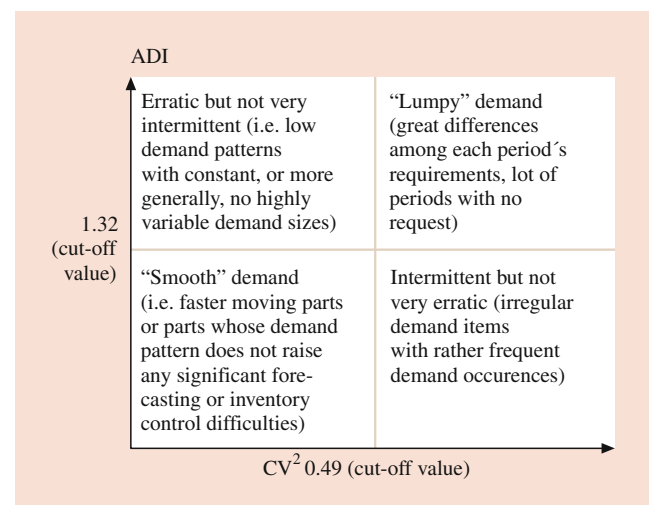


Fig. 15.1 Categorization of demand pattern

standard deviation of period requirements divided by the average period requirements.

Forecasting systems generally depend on the category of part used. Therefore, it is much more important to have two dependent factors to indicate deviation from expected values of demand, in respect of both demand size and inter-demand interval. The performance of a forecasting method should vary with the level and type of lumpiness. A classification of research on intermittent demand forecasting can be arranged according to Willemain as follows:

1. Extension of standard methods (Lau and Wang, Tyworth and O'Neill) [8, 9] and variants of Poisson model (Ward, Williams, Mitchell et al., Van Ness et al., Schultz, Watson, Dunsmuir et al.) [10–15]
2. Reliability theory and expert systems (Petrovic & Petrovic) [16]
3. Single exponential smoothing, Winter's models (Bier, Sani, and Kingsman, Willemain et al.) [17, 18]
4. Croston's variant of exponential smoothing (Croston, Segestedt, Willemain et al., Johnston) [19–21]
5. Bootstrap methods (Bookbinder and Lordahl, Wang and Rao, Kim et al., Park and Willemain) [22–25]
6. Moving average and variants (Ghobbar et al., Bartezzaghi) [26, 27]
7. Model based on binomial distribution (Pareschi et al.) [32–35]

The principle forecasting methods are briefly summarized in Table 15.1.

15.1.1 Exponential Smoothing

Exponential Smoothing methods (ES) are largely used time-series methods when reasonably good forecasts are needed over the short-term horizon, using historical data to obtain a “smoothed” value for the series. This smoothed value is then extrapolated to become the forecast for the future value of the series. The ES methods apply to past data an unequal set of weights exponentially decreasing with time, that is, the more recent the data value, the greater its weighting. In particular, the general form used in computing a forecast by the method of Single Exponential Smoothing (SES) is given by Eq. 15.2, where F_t represents the “smoothed estimate,” X_t the actual value at time t , and α the smoothing constant which has a value between 0 and 1. SES is best suited to data that exhibits a flat trend:

$$F_{t+1} = \alpha X_t + (1 - \alpha) F_t \quad (15.2)$$

When a trend exists, the forecasting technique must consider the trend as well as the series average; ignoring the trend will cause the forecast to underestimate or to overestimate,

according to an increasing or decreasing trend, the actual demand. In fact, Double Exponential Smoothing (DES), useful when the historic data series are not stationary, applies SES twice, and the general form is

$$F''_{t+1} = \alpha F_{t+1} + (1 - \alpha) F_t \quad (15.3)$$

15.1.2 Croston's Method

A little-known alternative to single exponential smoothing is Croston's method that separately forecasts the nonzero demand size and the inter-arrival time between successive demands using simple exponential smoothing (SES), with forecasts being updated only after demand occurrences. Let F_t and Y_t be the forecasts of the $(t + 1)$ th demand size and inter-arrival time, respectively, based on data up to demand t and Q_t the inter-arrival time between two successive nonzero demand. Then Croston's method gives:

$$F_t = (1 - \alpha) F_{t-1} + \alpha Y_t \quad (15.4)$$

$$Y_t = (1 - \alpha) Y_{t-1} + \alpha Q_t \quad (15.5)$$

The predicted demand at time t is the ratio between F_t and Y_t

$$P_t = F_t / Y_t \quad (15.6)$$

SES and Croston methods are most frequently used for low and intermittent demand forecasting; in particular, Croston can be useful with intermittent, erratic, and slow-moving demand, and its use is significantly superior to ES under intermittent demand conditions, accordingly with the categorization scheme of Fig. 15.1. Also, the straight Holt method (EWMA) is applicable only with low levels of lumpiness. The widespread use of SES and MTBR methods for parts with high variation (lumpy demand) is questionable as they consistently create poor forecasting performance which remains poor as the demand variability increases. The only method that fits quite well lumpy demand is the WMA, and its superiority to ES methods is proved: Its use could provide tangible benefits to maintenance service organizations forecasting intermittent demand. We mean for WMA the moving average method in which for computing the average of the most recent n data, the more recent observations are typically given more weight than older observations.

15.1.3 Holt Winter Models

Winter's models (AW, MW) methods consider the seasonal factor and provide the highest forecasting error with high

variation (lumpy demand). While computing Holt-Winter filtering of a given time series, unknown parameters are determined by minimizing the squared prediction error.

α , β , and γ are parameters of Holt-Winter filter, respectively, for level, trend, and seasonal component; if beta is set to 0, the function will perform exponential smoothing, while if gamma parameter is set to 0, a nonseasonal model is fitted.

The additive Holt-Winter prediction function (for time series with period length p) is

$$\bar{Y}[t+h] = a[t] + h \cdot b[t] + s[t+1+(h-1)|p|] \quad (15.7)$$

where $a[t]$, $b[t]$, and $s[t]$ are given by

$$a[t] = \alpha (\bar{Y}[t] - s[t-p]) + (1-\alpha) (a[t-1] + b[t-1]) \quad (15.8)$$

$$b[t] = \beta (a[t] - a[t-1]) + (1-\beta) b[t-1] \quad (15.9)$$

$$s[t] = \gamma (\bar{Y}[t] - a[t]) + (1-\gamma) s[t-p] \quad (15.10)$$

The multiplicative Holt-Winter prediction function (for time series with period length p) is

$$\bar{Y}[t+h] = (a[t] + h \cdot b[t]) \cdot s[t+1+(h-1)|p|] \quad (15.11)$$

where $a[t]$, $b[t]$, and $s[t]$ are given by

$$a[t] = \alpha \left(\frac{\bar{Y}[t]}{s[t-p]} \right) + (1-\alpha) (a[t-1] + b[t-1]) \quad (15.12)$$

$$b[t] = \beta (a[t] - a[t-1]) + (1-\beta) b[t-1] \quad (15.13)$$

$$s[t] = \gamma \left(\frac{\bar{Y}[t]}{a[t]} \right) + (1-\gamma) s[t-p] \quad (15.14)$$

The function tries to find the optimal values of α and/or β and/or γ by minimizing the squared one-step prediction error, if they are omitted. For seasonal models, starting values for a , b , and s are detected by performing a simple decomposition in trend and seasonal component using moving averages on first periods (a simple linear regression on the trend component is used for starting level and trend). For level/trend-models (no seasonal component), starting values for a and b are $X[2]$ and $X[2] - X[1]$, respectively. For level-only models (ordinary exponential smoothing), the start value for a is $X[1]$.

15.2 Forecasting Methods

In our opinion, considering major approaches to forecasting although many different classification schemes could be used, maybe the most significant scheme for their classification divides those methods into three major categories, as summarized in Table 15.2 [2]: judgmental, quantitative, and technological. Each category includes several types of methods, many individual techniques, and variations. *Judgmental methods* are most commonly used in business and government organizations. Such forecasts are most often made as individual judgments or by committee agreements. Nevertheless, quantitative methods are better than judgmental ones in determining spare part-inventory level, and we can suggest judgmental methods only in extremis.

The second category – *quantitative methods* – is the type on which the majority of the forecasting literature has been focused. There are three subcategories of these methods. *Time-series methods* seek to identify historical patterns (using a time reference) and then forecast using a time-based extrapolation of those patterns. *Explanatory methods* seek to identify the relationships that led to observed outcomes in the past and then forecast by applying those relationships to the future. *Monitoring methods*, which are not yet in widespread use, seek to identify changes in patterns and relationships. They are used primarily to indicate when extrapolation of past patterns or relationship is not appropriate.

The third category – *technological methods* – addresses long-term issues of a technological, societal, economic, or political nature. The four subcategories here are extrapolative (using historical patterns and relationships as a basis for forecasts), analogy-based (using historical and other analogies to make forecasts), expert-based, and normative-based (using objectives, goals, and desired outcomes as a basis for forecasting, thereby influencing future events).

15.2.1 Characterizing Forecasting Methods

In describing forecasting methods, we have found 7 most important factors reflecting their inherent capabilities and adaptability.

1. *Time horizon* – two aspects of the time horizon relate to individual forecasting method. First is the span of time in the future for which different forecasting methods are best suited. Generally speaking, qualitative methods of forecasting are used more for longer-term forecasts, whereas quantitative methods are used more with intermediate and shorter-term situations. The second important aspect of the time horizon is the number of periods for which a forecast is desired. Some techniques are appropriate for

Table 15.2 A summary of selected forecasting methods

No.	Method	Abbreviation	Description
1	Additive Winter	AW	Assumes that the seasonal effects are of constant size
2	Multiplicative Winter	MW	Assumes that the seasonal effects are proportional in size to the local de-seasonalized mean level
3	Seasonal regression model	SRM	It is used in time series for modeling data with seasonal effects
4	Component service life (replacement)	MTBR	Estimates of the service life characteristics of the part (MTBR = Mean Time Between Replacement), derived from historical data
5	Croston	Croston	Forecasting in case of low and intermittent demand
6	Single exponential smoothing	SES	Forecasting in case of low and intermittent demand
7	Exponential weighted moving average	EWMA, Holt	An effective forecasting tool for time series data that exhibits a linear trend
8	Trend adjusted exponential smoothing	TAES	Forecasting time series data that have a linear trend
9	Weighted moving averages	WMA	A simple variation on the moving average technique that allows for such a weighing to be assigned to the data being averaged
10	Double exponential smoothing	DES	Forecasting time series data that have a linear trend
11	Adaptive-response-rate single exponential smoothing	ARRSES	Has an advantage over SES in that it allows the value of α to be modified in a controlled manner as changes in the pattern of data occur
12	Poisson's model	POISSON	Models based on Poisson's distribution with the definition of a customer's service level
13	Binomial models	BM	Methods based on binomial distribution

forecasting only one or two periods in advance; others can be used for several periods. There are also approaches for combining forecasting horizons of different lengths.

2. *Pattern of data* – underlying the majority of forecasting methods is an assumption as to the type of pattern(s) found in the data to be forecast, for example, some data series may contain seasonal as well as trend patterns; others may consist simply of an average value with random fluctuations around that, and still others might be cyclical. Because different forecasting methods vary in their ability to predict different types of patterns, it is important to match the presumed pattern(s) in the data with the appropriate technique.
3. *Cost* – generally three direct elements of costs are involved in the application of a forecasting procedure: development, data preparation, and actual operation. The variation in costs obviously affects the attractiveness of different methods for different situations.
4. *Accuracy* – closely related to the level of detail required in a forecast is desired accuracy. For some decision situations, plus or minus 10% may be sufficient while in others a little variation of 2% could spell disaster.
5. *Intuitive appeal, simplicity, and ease of application* – a general principle in the application of scientific methods to management is that only methods that are understood in deep get used by decision makers over time. This is particularly true in the area of forecasting.
6. *Number of data required from past history* – some methods produce good results without consistent number of data from the past, because they are less affected by estimation errors in input parameters.
7. *Availability of computer software* – it is seldom possible to apply a given quantitative forecasting method unless

an appropriate computer program exists. Such a software must be user friendly and well conceived.

15.3 Forecasting Methods Applicability Related to Spare Parts Demands

Companies have to select in advance the appropriate forecasting method matching their cyclical demand for parts at best. Particular attention has to be paid to the demand for service parts inventories that is generally irregular and difficult to predict [3]. A summary of the better forecasting methods in relation to the categorization scheme of Fig. 15.1 is presented in Table 15.3.

15.4 Prediction of Aircraft Spare Parts: A Case Study

Airlines technical divisions are based on total hours flied and on the fleet size. With this data, the purchasing department tries to determine the quantity of stock necessary for a particular operating period. Alternatively, when new types of aircrafts are introduced, the airframe and engine manufacturers normally provide a recommended spares provisioning list, which is based on the projected annual flying hours, and includes forecast usage information on new aircraft. Also, the original equipment manufacturers provide overhaul manuals for aircraft components capable of supporting the assessment of required parts based on reliability information, i.e., on the specified components' operational and life limits. Consequently, the forecast of spare parts is practically based on past usage patterns and personnel experience.

Table 15.3 Summary of the better forecasting methods

Forecasting methods	Categorization of the demand			
	Intermittent	Erratic	Slow moving	Lumpy
Additive Winter – AW		•	•	
Multiplicative Winter – MW		•	•	
Seasonal regression model – SRM		•	•	
Component service life (replacement)		•	•	
Croston	•	•	•	
Single exponential smoothing – SES		•	•	
Double exponential smoothing – DES		•	•	
Exponentially weighted moving average – EWMA		•	•	
Trend-adjusted exponential smoothing		•	•	
Weighted moving averages	•	•	•	•
Adaptive-response-rate single exponential smoothing		•	•	
Poisson models	•		•	
Binomial models	•	•	•	•

Before any consideration about lumpiness and aircraft spare parts forecast, a discussion on the selection of the main variables used as “clock” for spare part life evaluation is absolutely necessary. According to Campbell’s study on maintenance records of United State air force [4], the spare parts demand appears strongly related to flying hours, but this sometimes does not appear the best indicator, e.g., to forecast landing gears, where it does not matter how long the aircraft is in the air, but how often it lands, or radar components that work only when the aircraft is on the ground. In conclusion, often flying hours are the best “clock,” but for each item a demonstration of its effectiveness is necessary [5].

In this study, different forecasting methods have been considered [6, 7], briefly:

- (a) Additive/Multiplicative Winter (AW/MW)

For each forecast, the optimal combination of level, trend, and seasonal parameters is realized. Available values for each variable (level and trend) are 0.2 and 0.01; seasonal length used is 12 periods.
- (b) Seasonal Regression Model (SRM)

Multiplicative model with trend and seasonal components. Seasonal length is 12 periods.
- (c) Single Exponential Smoothing (SES)

The statistical software applied (Minitab 14.0©) supports the research of the optimal weight of smoothing constant. Then the result is the best forecast with this method.
- (d) Double Exponential Smoothing (DES)

Dynamic estimates are calculated for two components: level and trend; the software supports their optimization. Also, in this case the best forecast with DES is guaranteed.
- (e) Moving Average (on the generic i -period) (MA(i))

Moving Averages are calculated with different time horizons (i period). The notation is MA(i).

This analysis employs every period length from 2 to 12.

- (f) Exponentially Weighted Moving Average (EWMA(i))

Also, in this case a weight optimization of smoothing coefficient for MA series has been realized. EWMA(i) is calculated for $i = 3, 4, 5$ and 8 periods.

Despite their importance in literature, we do not evaluate and compare methods based on Poisson approach because they reveal inadequate in intermittent demand prediction.

The case study deals with more than 3000 different items, with different levels of lumpiness, of Airbus fleet belonging to Italian national-flag airline. For each component, records relate to the daily demand level grouped in monthly interval of item usage, from 1998 to 2003. In terms of lumpiness, these avionic spare parts are classified in five different classes of behavior; for each class, a representative item, named $a, x, y, z,$ and w for confidentiality, is indicated.

Figure 15.2 presents an exemplifying demand of item z . The 5 “lumpiness” levels are reported in Fig. 15.3.

The Mean Absolute Deviation (MAD) of the forecast error is adopted as a performance index

$$MAD = \frac{\sum_{i=1}^n |\varepsilon_{fi} - \varepsilon_{fi}|}{n} \tag{15.15}$$

where ε_{fi} is the forecasted demand for period i . Some authors propose the Mean Absolute Percentage Error (MAPE) for this comparison, but in lumpy condition many item demands are zero, and as a consequence MAPE is not defined. For this reason, some authors propose a similar index, called MAD/A, also defined when the demand for items is zero:

$$MAD/A = \frac{MAD}{AVERAGE} \tag{15.16}$$

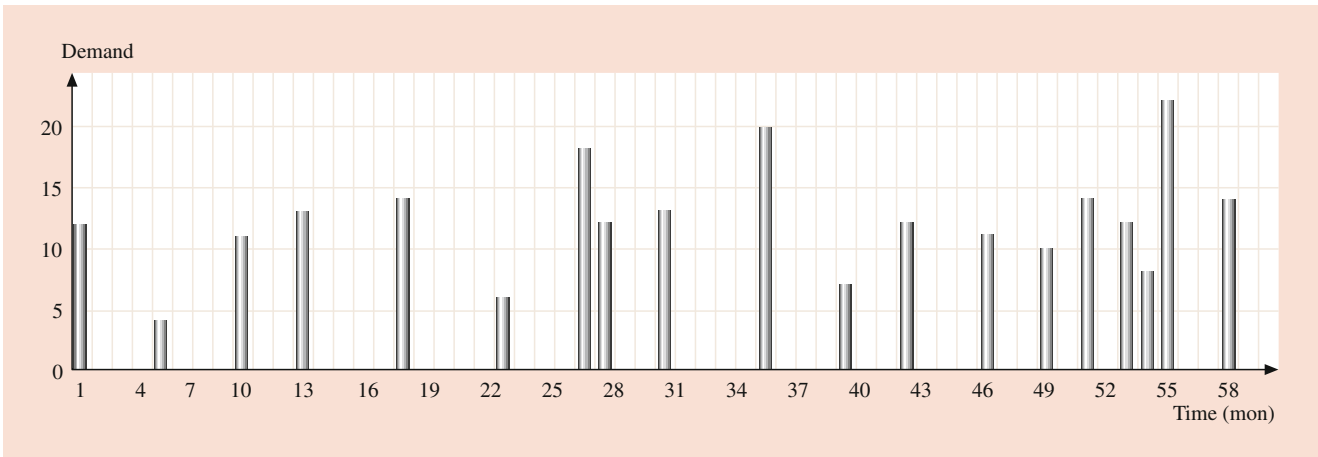


Fig. 15.2 Demand pattern for item z

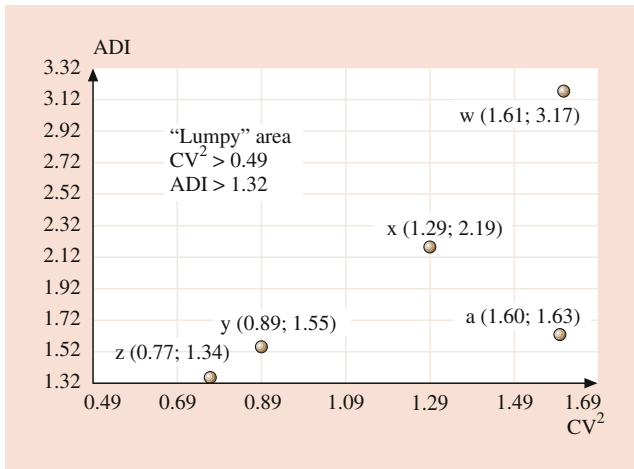


Fig. 15.3 CV² and ADI on monthly period for five representative "lumpy" items

where AVERAGE is the average value of historical item demand. Tracking signal (TS), as defined by Brown, is used to check if forecasts are in control or not.

$$TS_t = \left| \frac{CUSUM_t}{EMAD_t} \right| \tag{15.17}$$

where $CUSUM_t = (\varepsilon_{rt} - \varepsilon_{ft}) + CUSUM_{t-1}$ and $EMAD_t = \alpha|\varepsilon_{rt} - \varepsilon_{ft}| + (1 - \alpha) EMAD_{t-1}$

Limit values of TS and the optimal α value (0.25) are derived from Alstrom and Madsen approach. For analyzed items, forecasts are in control since the third year (that is their tracking signals respect the limits). The different forecasting methods are compared for all items and in particular for the proposed five components.

Table 15.4 and Figs. 15.4 and 15.5 show, respectively, some brief and full results of MAD and MAD/A for item z. Table 15.5 presents, for each representative item, the list of forecasting methods ordered by decreasing MAD, thus as-

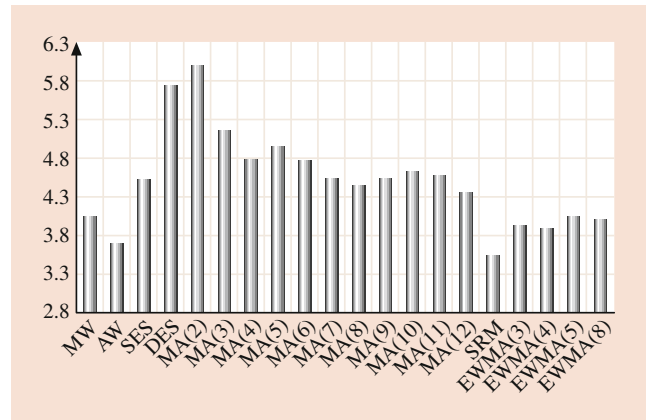


Fig. 15.4 MADs for item z

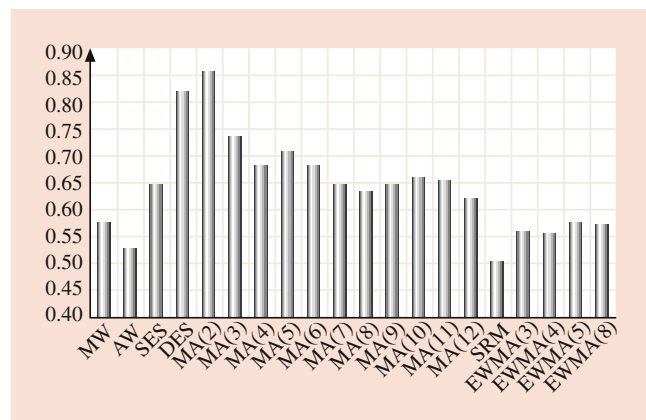


Fig. 15.5 MAD/As for item z

signing a "relative weight" related to the ranking position: A simple elaboration of these weights permits a full comparison in terms of total and average scores (* MW not defined for w item, due to its characteristics).

By means of MAD/A, it is possible to compare different forecasting methods on different items and their behavior in

Table 15.4 Comparison among some methods

Item z	MW	AW	SES	DES	MA(3)	MA(4)	MA(5)	MA(8)	SRM	EWMA(3)	EWMA(4)	EWMA(5)	EWMA(8)
MAD	4.04	3.71	4.54	5.74	5.16	4.80	4.97	4.43	3.53	3.92	3.88	4.06	4.01
MAD/A	0.58	0.53	0.65	0.82	0.74	0.68	0.71	0.63	0.50	0.56	0.55	0.58	0.57

Table 15.5 Ranking based on performance evaluation (MAD)

Weight	z	y	x	a	w	Method	Tot score	Average score
20	SRM	EWMA(3)	SRM	EWMA(4)	SRM	SRM	93	18,6
19	AW	SRM	AW	EWMA(3)	EWMA(5)	EWMA4	89	17,8
18	EWMA(4)	SES	MW	EWMA(5)	EWMA(4)	EWMA3	86	17,2
17	EWMA(3)	EWMA(4)	EWMA(5)	EWMA(8)	EWMA(3)	EWMA5	84	16,8
16	EWMA(8)	EWMA(5)	EWMA(4)	SES	SES	EWMA8	76	15,2
15	MW	EWMA(8)	SES	MW	AW	MW *	60 *	15
14	EWMA(5)	AW	EWMA(8)	SRM	EWMA(8)	SES	75	15
13	MA(12)	MA(10)	EWMA(3)	MA(7)	MA(5)	AW	74	14,8
12	MA(8)	MW	MA(10)	MA(8)	MA(4)	MA12	51	10,2
11	MA(7)	MA(11)	MA(11)	MA(11)	MA(12)	MA11	49	9,8
10	SES	MA(12)	MA(9)	MA(4)	MA(9)	MA9	47	9,4
9	MA(9)	MA(9)	MA(12)	MA(9)	MA(10)	MA10	47	9,4
8	MA(11)	MA(6)	MA(8)	MA(12)	MA(11)	MA7	44	8,8
7	MA(10)	MA(8)	MA(7)	AW	MA(7)	MA8	43	8,6
6	MA(6)	MA(7)	MA(5)	MA(10)	MA(8)	MA4	35	7
5	MA(4)	MA(5)	MA(6)	MA(6)	MA(6)	MA5	32	6,4
4	MA(5)	MA(4)	MA(4)	MA(5)	MA(3)	MA6	29	5,8
3	MA(3)	MA(3)	MA(3)	MA(3)	DES	MA3	16	3,2
2	DES	DES	MA(2)	DES	MA(2)	DES	10	2
1	MA(2)	MA(2)	DES	MA(2)		MA2	7	1,4

face of different lumpiness conditions: SRM, EWMA(i), and Winter are the best forecasting methods. This result is not related to lumpiness level, at least for lumpiness represented by $ADI < 3.3$ and $CV^2 < 1.8$, typical range for aircraft components. Some interesting observations can be drawn:

- Figure 15.6 clearly attests the item lumpiness as a dominant parameter, while the choice of the forecasting method has a secondary relevance; all methods for a “little-lumpy” item (e.g., items y and z) generally perform better than the best method for a “high-lumpy” component (e.g., items x and w). However, lumpiness is an independent variable and is not controllable.
- The value of the average MAD/A calculated for all forecasts generated by all methods is 1.02. The aim of this study is a comparison of different forecasting methods, but we can conclude that demand forecasting for lumpy items is very difficult, and results are not very accurate. Besides, lumpy demand is often equal to zero or one: All prediction lower than 1 must be rounded up to one. This phenomenon introduces another source of error.

- For a single component, the average fluctuation (in terms of MAD/A) of the ratio maximum/minimum, among different techniques, is about 1.55 (usually between 1.40 and 1.70); for a single forecasting method, the average fluctuation (in terms of MAD/A) of the ratio maximum/minimum, among different components, is about 2.17 (usually between 1.57 and 2.18). Then the main relevance of lumpiness is proclaimed again.
- Analyzing the effectiveness of a single model, research demonstrates (Tables 15.3 and 15.4) that the Seasonal Regression Model (SRM), the Exponentially Weighted Moving Average (EWMA(i)), and Winter are the best forecasting methods. It is important to remember that analyzed items are effectively representative of a population of aircraft spare parts. This result is not related to lumpiness level, at least for lumpiness represented by $ADI < 3.3$ and $CV^2 < 1.8$ (typical range for aircraft components).

In conclusion, intermittent demand for service parts, usually highly priced, is a very critical issue, especially for lumpy demand prediction, as typical for avionic spare parts.

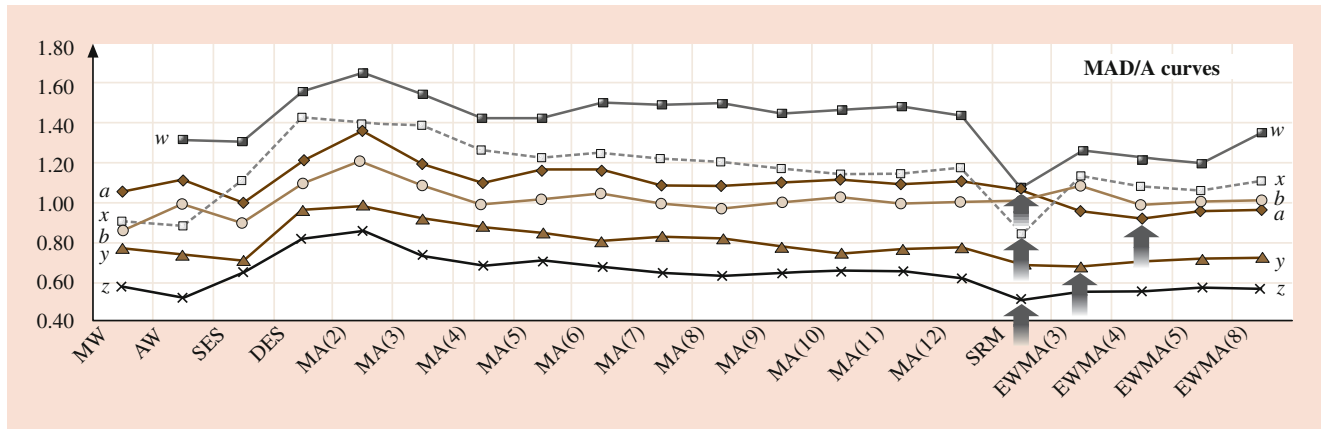


Fig. 15.6 MAD/A data and curves

In literature, lumpy demand forecasting is not deeply investigated, Ghobbar interesting research apart, and sometimes conflicting results are recovered. The introduction of economic question is the final development: It is absolutely necessary to check the impact of stocking costs and costs due to component lacks on the forecasting methods; aircraft operator can incur costs of more than 30,000 \$ per hour if a plane is on the ground.

15.5 Poisson Models

For builders of high-technology products, as automatic packaging machines, the supply of spare parts creates a strategic advantage respect to the competitors, with particular regard to Low Turnaround Index (LTI) parts conceived and designed for the satisfaction of a specific customer request. The strategic problem to solve is to individualize the minimum number of spare parts capable of avoiding the customers plants' downtimes for a specific period, called covering period, that coincides with the time between two consecutive consignments.

The procedure actually used by a great number of manufacturers, named "recommended parts," consists in the creation, at design stage, of some different groups of replaceable parts with different covering time for every functional machine group. This methodology is very qualitative, and it strongly depends on the designer opinion; besides, not considering the information feedback from customers, it usually estimates in excess the number of spare parts in respect to the real demands of customers. Even though this way avoids plant downtimes, absolutely forbidden for the high costs of production loss, it normally creates exuberant and expensive stocks, with unsought high risks of damage and obsolescence. For LTI items, the usual economic batch or safety stock methods are not suitable to forecast the spare

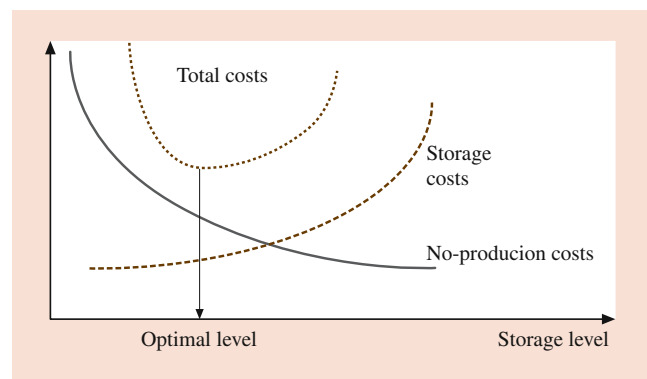


Fig. 15.7 Economic approach

parts amount. For such a situation, during last years a lot of different approaches, usually based on Poisson distribution, have been developed: Stock level conditioned to minimal availability or to minimum total cost (see Fig. 15.7) is considered the most interesting.

Every study reported in literature assumes that item's failure time (for spare parts demand) is exponentially distributed and as a consequence the failure rate $\lambda(t)$ is independent from the time. This simplifying hypothesis is due to the difficulties in estimating Mean Time Before Failure (MTBF) real values. Finally, it is important to underline that the quantity of spare parts and its temporal distribution also represent a strategic information during the negotiations with customers for the purchase of the plants and the quantification of related costs.

15.5.1 Stock Level Conditioned to Minimal Availability Method

This method first needs to calculate the asymptotic availability A by the known formula:

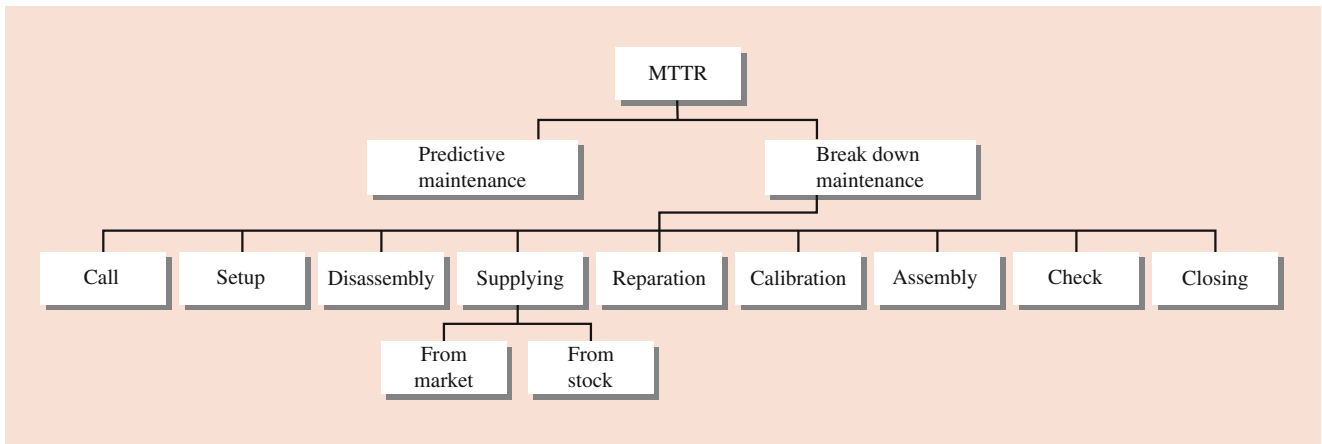


Fig. 15.8 MTTR’s structure

$$A = \frac{MTBF}{MTBF + MTTR} \tag{15.18}$$

The Mean Time To Repair (MTTR) term is derived from different factors, as shown in Fig. 15.8:

Its value strongly depends on the spare part being on consignment or not, and it can be calculated by the formula:

$$MTTR = T_1 + \int_0^{T_s} (T_s - T_x) f(T_x) dT_x = MTTR(N) \tag{15.19}$$

where T_1 is amount of time due to factors except time of supply (for instance, disassembling), T_s lead-time of supply for not available components, N number of spare parts available in stock at the time zero, T_x time interval between the instant when the consumption of the part reaches the value N (empty stock situation) and the consignment of spare part, and $f(t)$ density of failure distribution.

$$P_N = \frac{(\lambda T_s)^N \cdot e^{-\lambda T_s}}{N!} \tag{15.20}$$

It is worth to note that for increasing N we get decreasing MTTR, increasing availability A , and the downtimes costs fall down. Second, the method affords the quantitative definition of the storage cost that requires the definition of the average number of parts storing during the time of supply T_s . If the warehouse contains N parts at time zero, the probability P_N of N failures in T_s can be pointed out by Poisson formula:

In the same way, it is possible to calculate the probability of one, two, or N failures.

Let R indicate the cost of each spare part, and s the stoking cost index per year; the annual stockcosts C can be evaluated by the formula $C = R \cdot s [N \cdot P_0 + (N - 1) \cdot P_1 + (N - 2) \cdot P_2 + \dots P_{N-1}]$ that can be used in an iterative process in order to find the optimum level N that leads to a minimum for the cost C , all the while allowing the minimum level of

availability $A_{MIN}(N)$ to guarantee on-time technical requests to be satisfied (for example, safety questions or productivity level).

$$\begin{cases} \min [C = R \cdot s (NP_0 + (N - 1) P_1 + (N - 2) P_2 + \dots + P_{N-1})] \\ \text{subject to } A(N) = \frac{MTBF}{MTBF + MTTR(N)} \geq A \quad \min \end{cases} \tag{15.21}$$

15.5.2 Stock Level Conditioned to Minimum Total Cost

The aim of this method is to determinate the total amount N of replaceable parts in order to minimize the total cost function C_{tot} defined by

$$C_{tot}(N) = C_1 + C_2 \tag{15.22}$$

The warehousing cost term C_1 can be estimated as in (15.21), while for the cost C_2 it is necessary to quantify the probability of stock-out situations. During the time T_s , production losses could occur if the number of failures exceeds the number N of parts supplied at the consignment time, assumed as zero. The cumulative probability calculated by Poisson distribution is:

$$P = P_{N+1} + P_{N+2} + P_{N+3} + \dots \tag{15.23}$$

Let d indicate the annual part consumption of a customer and CM the cost corresponding to a loss of production; the term due to stock-out is

$$C_2 = CM \cdot d \cdot P \tag{15.24}$$

For a rapid choice, it is possible to employ the following user-friendly abacus (Fig. 15.9):

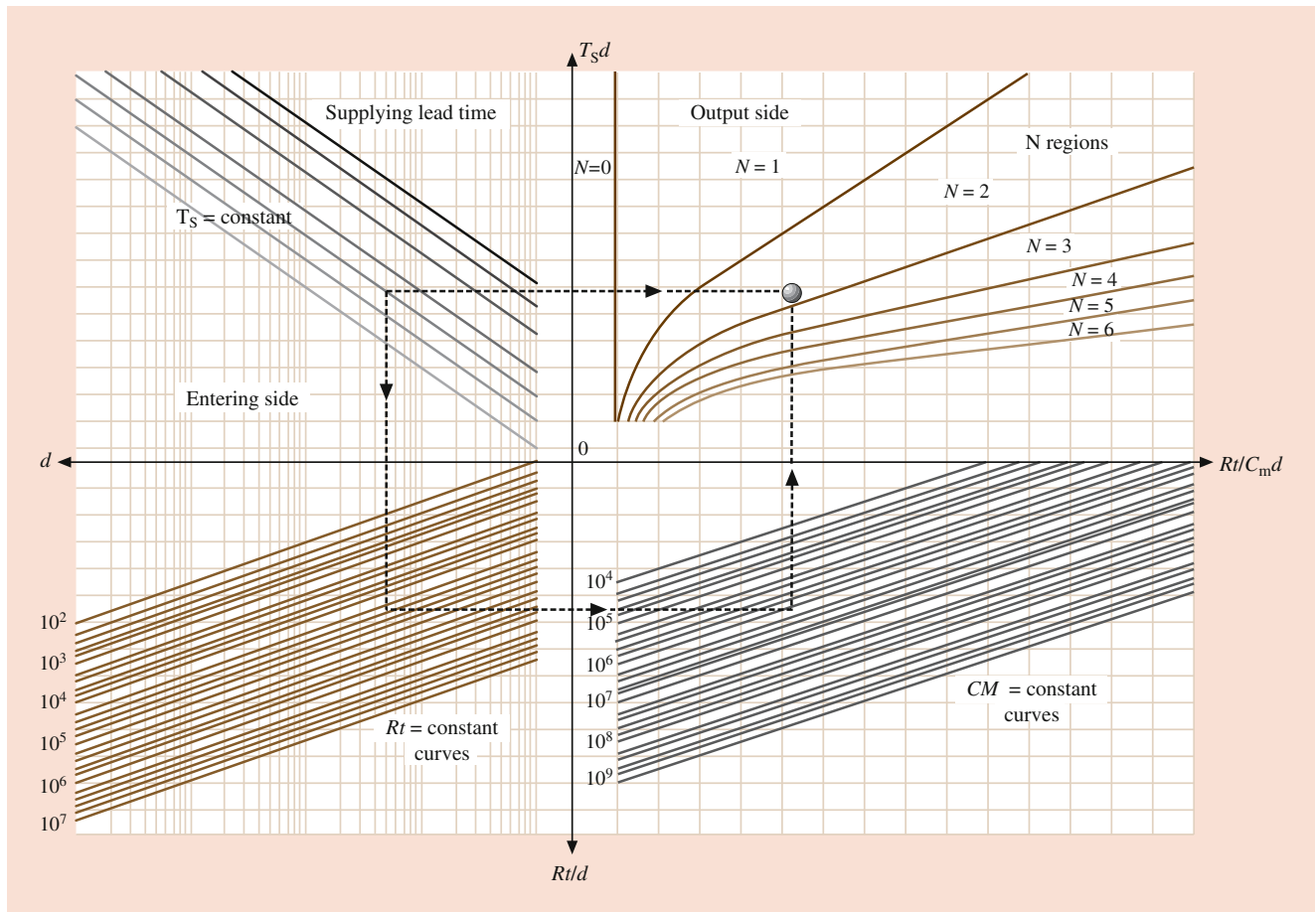


Fig. 15.9 Graphic abacus

15.6 Models Based on Binomial Distribution

Industrial applications show that the methodologies based on Poisson formula usually overestimate the real replacements consumption. To overcome this problem, we present a new quantitative procedure that does not consider the request of parts being linear during time, as Poisson procedure does.

The innovative approach calculates the requirement of components, for a given covering period T , by the addition of two addenda x_1 and x_2 : The first is related to the wear damage of the replaceable component and can be deducted by MTBF value, while the second is related to the randomness of the breakdowns and covers the possibility of failures in advance respect to the average situation. The optimal number of replacements is $N = x_1 + x_2$.

Let n be the number of different employments for component in several machines owned by the customer and T the covering period; x_1 can be expressed by:

$$x_1 = \text{int} \left(\frac{T}{\text{MTBF}} \right) \cdot n \tag{15.25}$$

This average term assumes interesting values only in presence of a high consumption of the component, in particular in the rare situations when an LTI part has a lot of applications, indicated by n . Anyway, this term x_1 represents a scanty information; we have to consider the second term that corresponds to the number of parts to get the required value of the customer service level LS in the residual time T_{residual} , defined as the residue of the ratio between T and MTBF. The customer service level is the probability that the customer finds the parts during the remaining period and can be fixed separately according to strategic and economic assessments.

The value of x_2 is got by the following formulas:

$$T_{\text{residual}} = T - \text{int} \left(\frac{T}{\text{MTBF}} \right) \cdot \text{MTBF} \tag{15.26}$$

$$p = Q(T_{\text{residual}}) = 1 - e^{-\left(\frac{T_{\text{residual}}}{\text{MTBF}}\right)} \quad (15.27)$$

where p represents the failure probability during T_{residual} ; by p and the binomial distribution, it is easy to calculate the probability that a component (with n applications) requires less than x_2 replacements in T_{residual} :

$$P(x \leq x_2; n; Q(T_{\text{residual}})) = \sum_{i=0}^{x_2} \binom{n}{i} (1-p)^{n-i} \cdot p^i \quad (15.28)$$

As a consequence, it is possible to quantify the no-stock-out confidence level to compare with the customer satisfaction as

$$\text{LS}(x_2) = 1 - P(x \leq x_2; n; 0Q(T_{\text{residual}})) \quad (15.29)$$

The main innovative result is that the procedure does not consider the total request of spare parts to be linear versus time, as other methods do; it tries to set the best moment for the supply in order to maximize the customer service level without increasing the average amount of spare parts. In fact, the new method respects the average consumption by the term x_1 and increases the customer service levels planning the requirement for spare parts in the residual time by the term x_2 .

15.6.1 An Industrial Application

This procedure is successfully running on PC systems in an Italian company leader in manufacturing of packaging machines. The supply of spare parts creates a strategic advantage in respect to competitors, because the automatic packaging machines usually present a long-life cycle and contain a lot of functional groups, often ad hoc conceived and designed. The economic impact of replacement activity is not negligible: It usually amounts up to 15% of the global business volume. Surely a good forecast of spares parts can simplify the manufacturer production planning. Before its industrial real-time application, the innovative procedure was tested to forecast the consumption of 190 different spare parts indicated by several customers over a 33-month period (Figs. 15.10 and 15.11).

The experimental results of these procedures were evaluated by two performance indexes:

1. Percentage number of spare parts without stock-out periods
2. Effective utilization of replacements by the customer

As a first test, the new procedure was applied with the restrictive hypothesis of a covering period of 1 month: Using LS equal to 99.8%, the method was good performing for 166 replaceable parts and there was a very good correlation



Fig. 15.10 A packaging machine example for cigarettes production

Fig. 15.11 Example of N evaluation for a specific item (code 0X931: pin for hoes gear levers)

INPUT	
MTBF (h)	3945
Positions (n)	26
Confidence level	97%
Supplying time (h)	2300

OUTPUT	
Failure ratio λ	0,000253
X_1	0
Tresidual (h)	2300
P(Tresidual)	0,442
X_2	See table
$N = X_1 + X_2$	16

X_2 (items)	LS	(X_1+X_2) (items)	X_2 (items)	LS	(X_1+X_2) (items)
0	0	0	14	0,883	14
1	0	1	15	0,943	15
2	0	2	16	0,976	16
3	0	3	17	0,991	17
4	0,002	4	18	0,997	18
5	0,007	5	19	0,999	19
6	0,022	6	20	1	20
7	0,055	7	21	1	21
8	0,118	8	22	1	22
9	0,218	9	23	1	23
10	0,351	10	24	1	24
11	0,505	11	25	1	25
12	0,658	12	26	1	26
13	0,787	13			

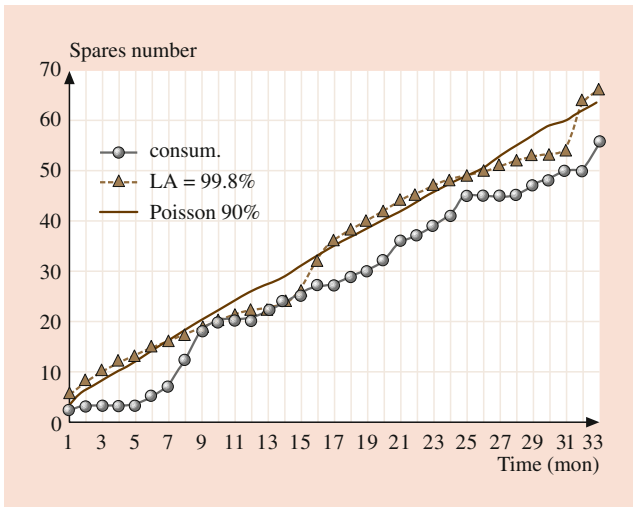


Fig. 15.12 Pin for hoes gear levers forecasts and applications

between prevision and real customer consumption, as shown in comparison in Fig. 15.12 to the real consumption and Poisson linear forecast (LS = 90%) for a pin for hoes gear levers with 26 applications in the customer machine park.

The 88% of components investigated did not present any month with stock-out, 3% presented less than 3 months of underevaluation, and 9% had more than 3 months in stock-out. On the other hand, the utilization index showed that the 87% of the components had a normal or good or optimal customer use: In other words, they did not remain in spare parts warehouse more than 15 days before the installation (see Fig. 15.13).

Moreover, parts with bad forecasts were investigated to understand the reasons with very encouraging results: In fact, errors were usually caused by preventive maintenance operations, machines revision, changes of suppliers, or changes

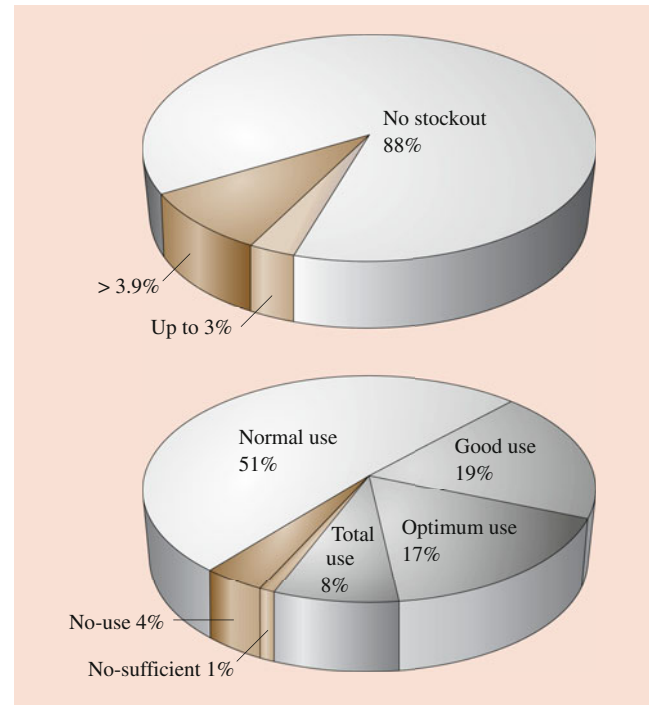


Fig. 15.13 Simulated stock-out periods (months) and real utilization of dispatched replacements

in the application of the component not pointed out by the customer. The effect of the extension of the covering period was analyzed by testing the new method with different values of T . For increasing T , we get an increasing stock of spare parts, but the number of stock breakages strongly decreases: In fact for $T = 1$ month, the method well-performed for the 88% of the parts is investigated, and this percentage increased up to 90% for $T = 3$ months, to 95% for $T = 6$ months, and to 98% for $T = 1$ year. Therefore, it was possible to study the

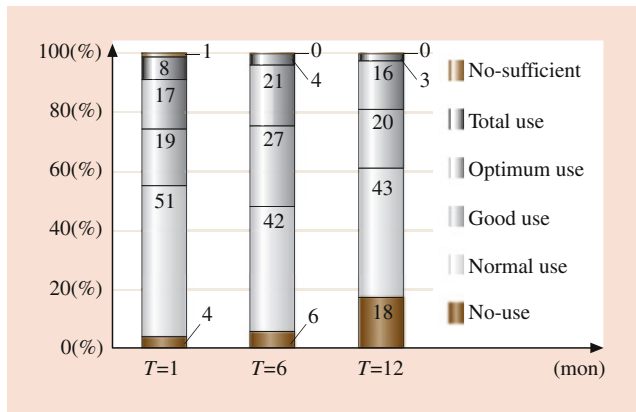


Fig. 15.14 Real use compared to supplying time T

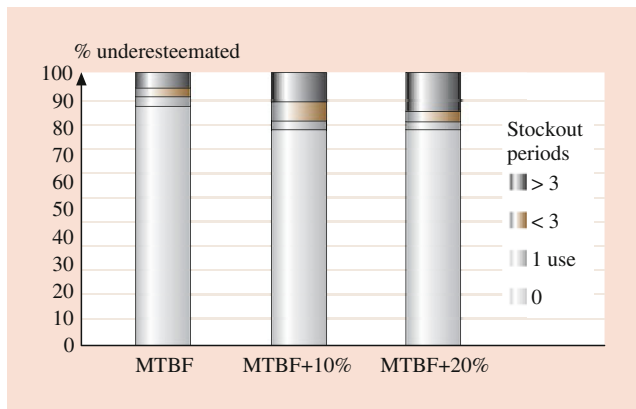


Fig. 15.15 Sensibility to MTBF evaluation

optimum extension of the covering period that for the 190 components investigated was found equal to 3 periods (see Fig. 15.14).

Some simulations with different values of MTBF show the influence of its approximate evaluation: Values of MTBF overestimated for 10% and 20% reduce the performance of the method, respectively, of 6% and 9% in terms of percentage number of spare parts without stock-out periods (see Fig. 15.15).

In spite of this important conclusion, it is important to remember that MTBF values have to be updated, starting from the initial value of $MTBF_{initial}$, by the feedback information from the customers; the most suitable control parameter is the component quantity Y employed by the customer during the covering period T and the relation (15.30) that gave the best results, as shown in Figs. 15.16 and 15.17:

$$MTBF_{updated} = \frac{MTBF_{initial} \cdot (MTBF_{initial}) + \frac{nT}{Y} \cdot (nT)}{MTBF_{initial} + nT} \quad (15.30)$$

where $MTBF_{updated}$ is the weighted average of $MTBF_{initial}$, and $\frac{nT}{Y}$ (weights under round brackets).

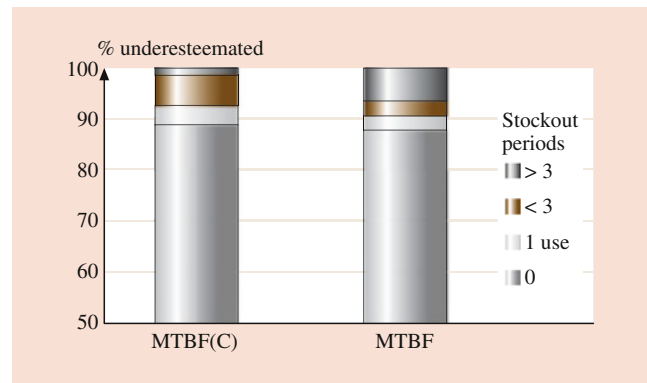


Fig. 15.16 MTBF correction effect on stock-out

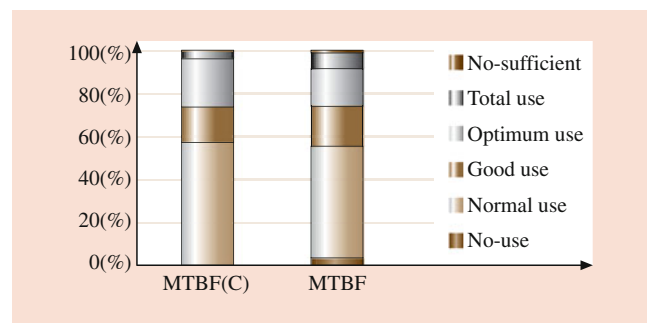


Fig. 15.17 MTBF correction effect on items' utilization

15.7 Binomial Model Extension Based on Total Cost Function

The proposed model required the assumption of a specific spare part LS defined as the probability of finding the part in case of breakdown. Some simulations with different values of LS show that it is important to assume $LS \leq 80\%$ and to reserve $LS \geq 90\%$ for particular situation, e.g., customers placed in a far country or without skilled workers. It is possible to determinate the number N of replaceable parts needs and therefore the LS value capable of minimizing a total cost function defined by the sum of storage costs and production losses costs [30, 31].

15.7.1 Service Level Optimization: Minimum Total Cost Method

The aim of this paragraph is to determine the requirement N of replaceable parts capable of minimizing a total cost function defined by the sum of production losses costs C_1 and storage costs C_2 . During the time T_s , production losses could occur if the number of failures exceeds the number N of supplied parts which are available after the consignment

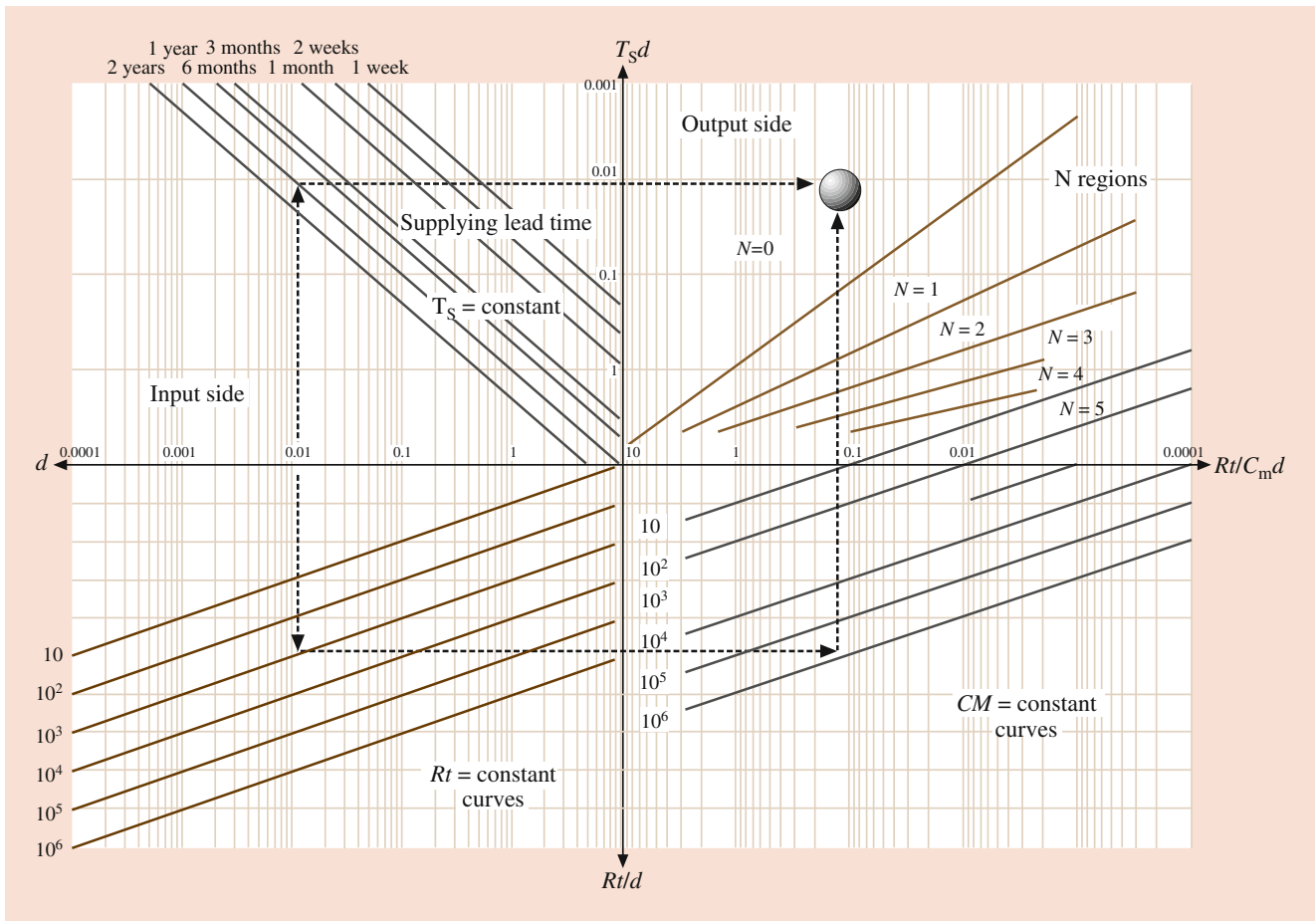


Fig. 15.18 Graphical solution for this methodology

Table 15.7 LS % and minimum cost related to $T_s d$ and $Rt/(C_m d) - n^{\circ}$ of employments $n = 15$

$Rt/(C_m d)$	LS % ($n = 5$)								
	$T_s d$	5	4	2	1	0.8	0.2	0.02	0.02
9	4.7	10.2	13.5	37	45.0	81.9	98	99.8	99.8
3	34.8	28.2	42.5	75	45.0	81.9	98	99.8	99.8
1.5	57.3	52	71.4	75	81.9	81.9	98	99.8	99.8
0.9	76.9	73.8	71.4	75	81.9	81.9	98	99.8	99.8
0.3	89.8	88.5	89.3	93	96	98.4	98	99.8	99.8
0.1	96.4	95.9	96.9	99	99.4	98.4	98	99.8	99.8
0.03	99	98.8	99.3	100	99.4	99.9	98	99.8	99.8
0.003	100	100	100	100	99.9	99.9	100	100	100
0.0003	100	100	100	100	100	100	100	100	100

internal use and for replacements ordered by customers according to a fixed EOQ (Economic Order Quantity). The application deals with the optimization of supplying time in order to reduce the total management costs of spare parts at assigned EOQ, that is, the aim is to define the time between consignments capable of reducing total costs with the same value of EOQ. Three components (a support grid, the

clamp, and a special gasket) are considered with a downtime calculated as

$$\text{Downtime} = \text{MTTR} \cdot d \cdot n \quad (15.37)$$

In particular, the support grid has $n = 5$ employments, $H = 1760$ total hours per year, $d = 0.5$ annual part consumption [unit/year] on the whole, $\text{MTTR} = 10$ [hours], $\text{MTBF} = 8800$ [hours], down time cost per hour $C_m = 1000$ [€/hour], and as a consequence, $\text{CM} = C_m \cdot \text{MTTR} = 10,000$ [€] cost for a production lack. This component was supplied in fixed EOQ with $N = 3$ elements, at a cost per unit $R = 100$ [€/unit] ($R \ll C_m$); the relative risk of damage and obsolescence suggests $t = 0,1$ for the annual stoking cost index. Entering the abacus with these values for N , CM , and $R \cdot t$ (see Fig. 15.18), we obtain as a result the optimal time for supply $T_s = 400$ [days]; by means of Eqs. (15.31) and (15.32) with $N = x_1 + x_2 = 0 + 3$ ($x_1 = 0$ because $\text{MTBF} < T_s$), we determine $\text{LS} \approx 99\%$: This very high level is due to low values for t and R/C_m rate.

The results following by the use of the graphic abacus are shown in Table 15.8 for the whole set of components.

Table 15.8 Optimization of T_S for fixed number of spare parts N

Component	$N = x_1 + x_2$	T_S (days)	LS
Support grid	3	400	99%
Clamp	4	90	98%
Special gasket	5	90	95%

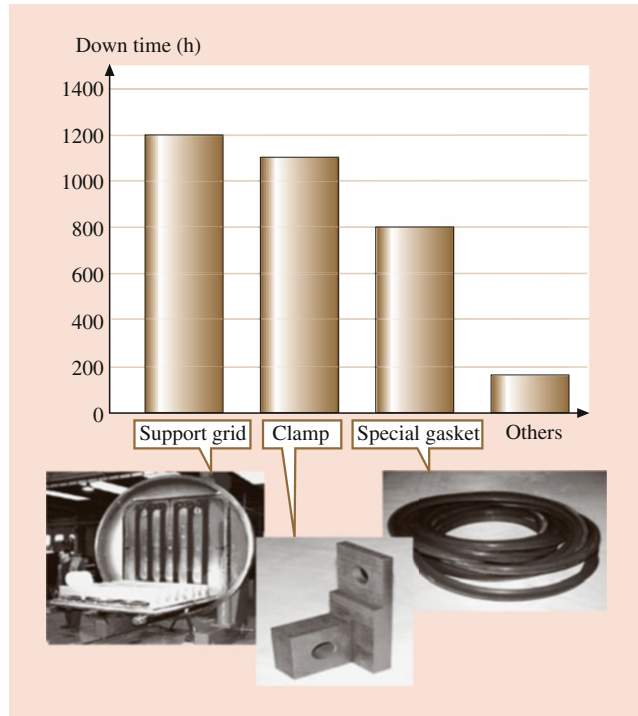


Fig. 15.19 Downtimes for grid, clamp, gasket, and all the other components (cumulative)

Results are summarized in Fig. 15.19 and compared with the output of an existent minimum cost method based on Poisson distribution.

15.8 Weibull Extension

The innovative methodology can be extended to the whole lifetime by implementing the Weibull failure rate function to LTI spare parts stock level in maintenance system with declared wear conditions. Weibull distribution is one of the most commonly used lifetime distributions, and it is flexible in modeling failure time data, as the corresponding failure rate function can vary or assumed to be constant. Literature offers a lot of papers dealing with models for bathtub-shaped failure rate. For example, Hjorth [36] proposed a three-parameter distribution; Mudholkar and Srivastava introduced an exponential Weibull distribution [37]; Chen [39] spoke about a two-parameter lifetime distribution with bathtub shape or increasing failure rate function; Xie [39] wrote a very interesting paper about a model that can be seen as a

generalization of the Weibull distribution and tries to improve the procedure of estimation of the parameters.

The estimation of well-known parameters η (scale) and β (shape) in a Weibull distribution can be performed graphically, but this is not accurate unless we have a large sample size, not always available in case of LTI spare parts; anyway, we focus our attention to the final zone in the traditional “bathtub” wear model, and our aim is to understand whether the hypothesis of constant failure rate in our previous works increases or not the spare parts costs, in comparison with more sophisticated distributions. For this reason, we developed our model by the use of traditional Weibull distribution, but any model mentioned above can be easily stretched.

15.8.1 The Modified Model Extension by Weibull Distribution

By historical data, it is possible to determine the cumulative percentage of components failures related to the lifetime. The graphic approach of Fig. 15.20 permits the definition of the Weibull distribution parameters η (scale parameter) and β (shape parameter). This is possible by Plait transformation (Weibull transformation): Starting from the failure rate calculated as in (15.38), the reliability as in (15.39), the cumulative distribution functions as in (15.40), and the definition of a normalized parameter x as in (15.41), find a linear correlation between the parameter x and cumulative distribution function (15.42 and 15.43), represented in Fig. 15.20.

$$\lambda(t) = \frac{d}{dt} \left(\frac{t}{\eta} \right)^\beta = \frac{\beta}{\eta} \left(\frac{t}{\eta} \right)^{\beta-1} = \frac{\beta}{\eta^\beta} t^{\beta-1} \quad (15.38)$$

$$R(t) = e^{-\left(\frac{t}{\eta}\right)^\beta} \quad (15.39)$$

$$F(t) = 1 - R(t) = e^{-\left(\frac{t}{\eta}\right)^\beta} \quad (15.40)$$

$$x = -\frac{t}{\eta} \quad (15.41)$$

$$\beta \text{Ln}(x) = \text{Ln} \left(\text{Ln} \left(\frac{1}{1 - F(x)} \right) \right) \quad (15.42)$$

The optimal replacements number of LTI spare parts are also given by relation (15.25), while relations (15.27) and (15.28) are modified with Weibull parameters η and β :

$$p = Q(T_{\text{residual}}) = 1 - e^{-\left(\frac{t}{\eta}\right)^\beta} \quad (15.43)$$

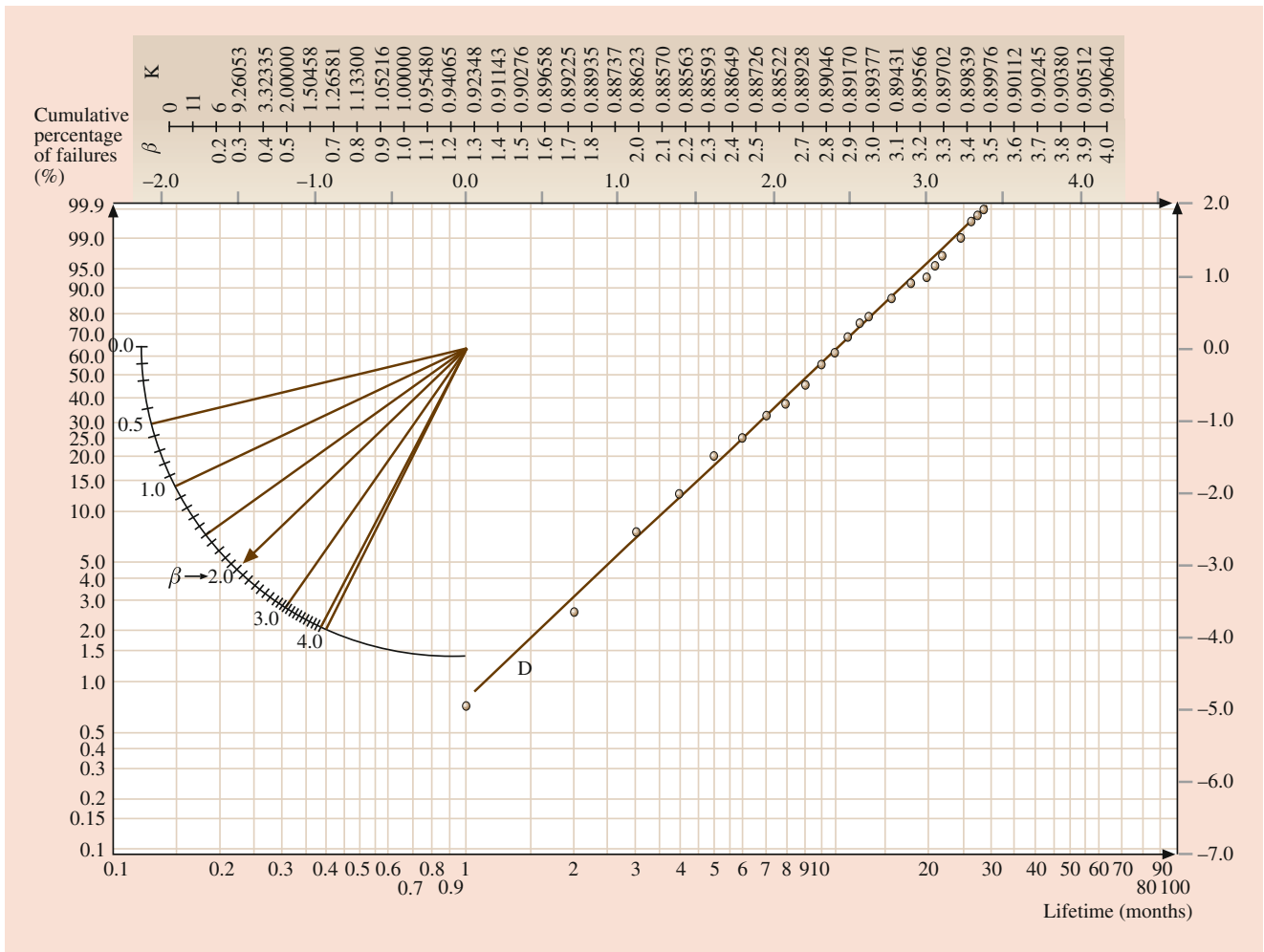


Fig. 15.20 Graphical estimation of the β value

$$P(x \leq x_2; n; Q(T_{\text{residual}})) = \sum_{i=0}^{x_2} \binom{n}{i} (1-p)^{n-i} \cdot p^i \tag{15.44}$$

It is possible to quantify the no-stock-out confidence level to compare with the customer satisfaction as $LS(x_2) = 1 - P(x \leq x_2; n; Q(T_{\text{residual}}))$. As previously stated, LS must be close to 100% when $C_m \gg R$, while in the opposite case the optimum LS is variable in function of $T_S \cdot d$, but in this case for assigned MTBF the optimum LS increases likewise the number of employments n .

15.8.2 Simulation and Results

This extended model is compared with previously proposed models for different values of the parameters involved. The relation between the parameters MTBF, T_S , and β appears

very interesting. In fact, the first two parameters are fundamental to find the quantity x_2 (15.29), (15.44), while β indicates the gap from the hypothesis of constant failure rate. Surface of Fig. 15.21 (with T_S equal to 500 hours and customer service level 95%) relates the difference of optimal replacement numbers, respectively, calculated by (15.29) or (15.44) for given MTBF and β values. For a fixed value of MTBF, the saving increases with greater values of β , because the use of Weibull distribution takes into account that the failures are grouped in a specific time region where part breakdowns present a high probability to occur. The MTBF value of 500 hours is very important because it defines T_{residual} equal to zero, and so x_2 is equal to zero for any approach. Values of MTBF lower than 500 hours mean that optimal replacements number is influenced by quantity x_1 , while values greater than 500 are defined by the only use of quantity x_2 (x_1 equal to zero). Considering an MTBF range starting from T_S value and for a specific value of β , the saving of spare parts needed decreases with greater values of MTBF,

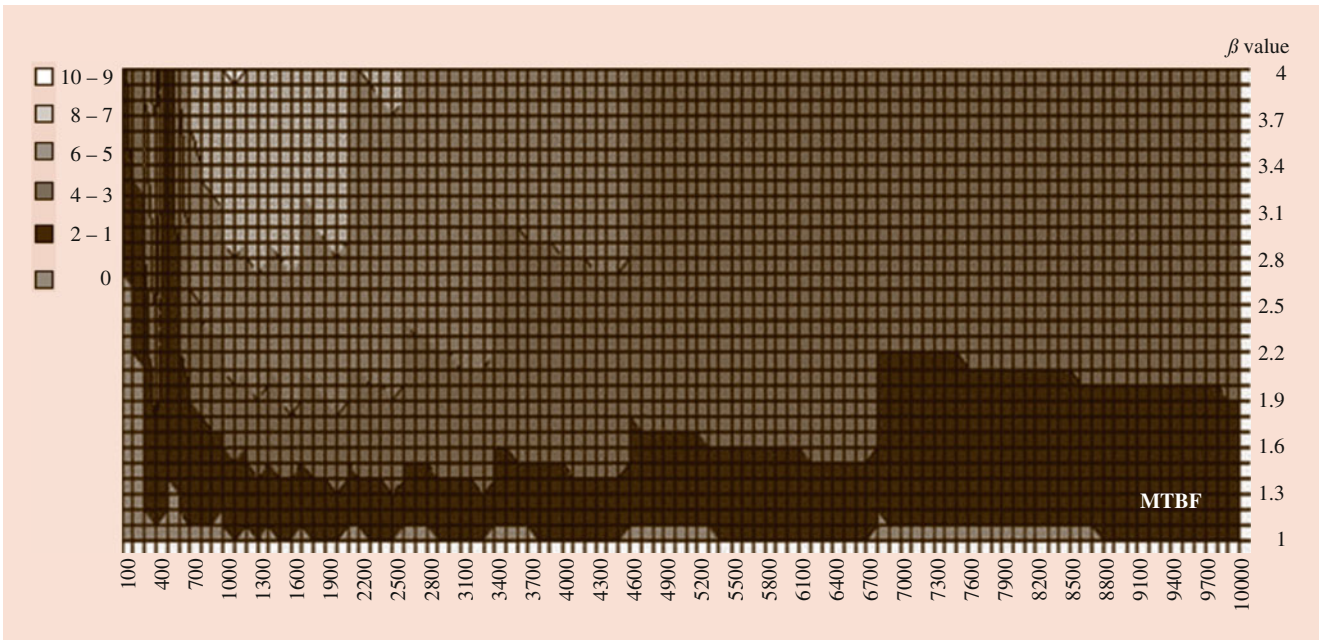


Fig. 15.21 Difference of optimal replacements numbers respectively calculated by (15.29) or (15.44) respect to the average life of the component MTBF and β values

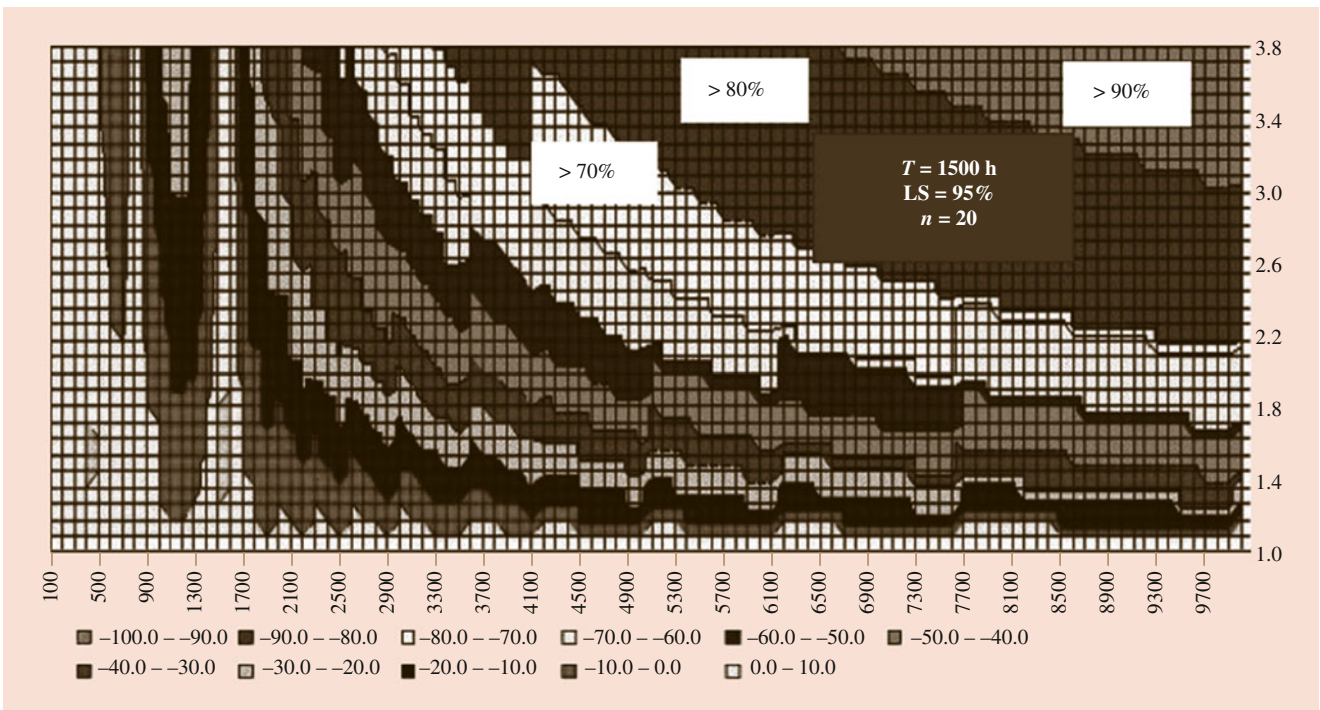


Fig. 15.22 Share of saving respect failure rate constant hypothesis in case of different values of parameters

as shown in Fig. 15.22 where two different values of the parameter T_S (respectively, 500 and 1000 hours) are compared. Figure 15.23, as Fig. 15.24, relates the difference between the optimal replacements number calculated by (15.29) and by (15.44) for T_S equal to 1500 hours.

Obviously, a low frequency of consignment creates a greater requirement of spare parts. Also in this case, it is important to notice that methods behave as one for MTBF equal to T_S .

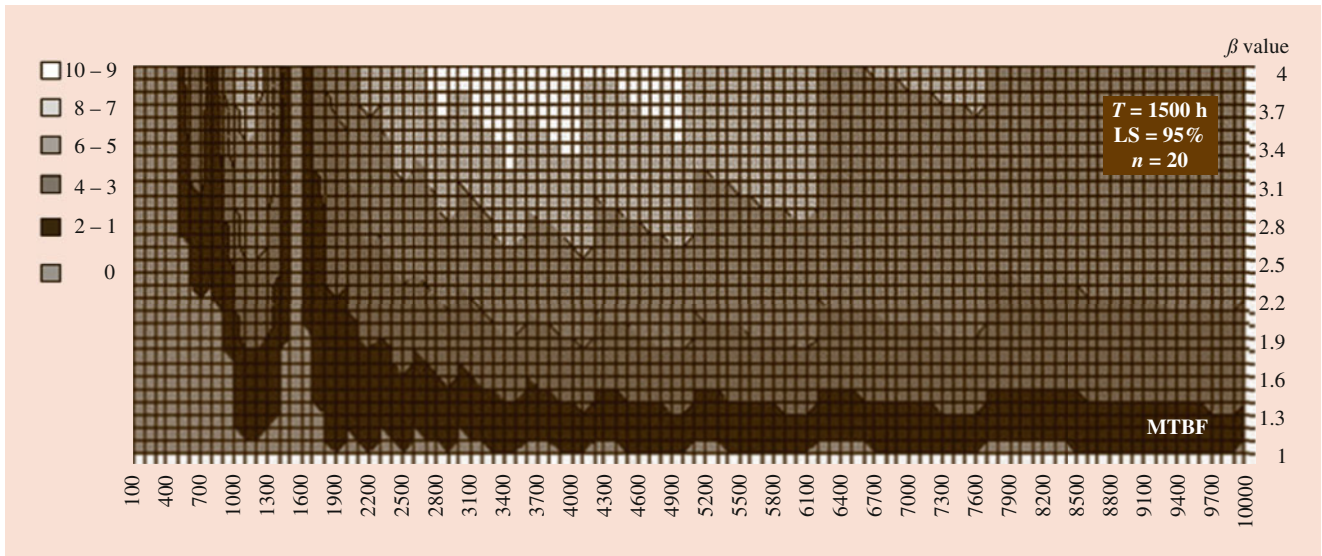


Fig. 15.23 Share of saving respect failure rate constant hypothesis

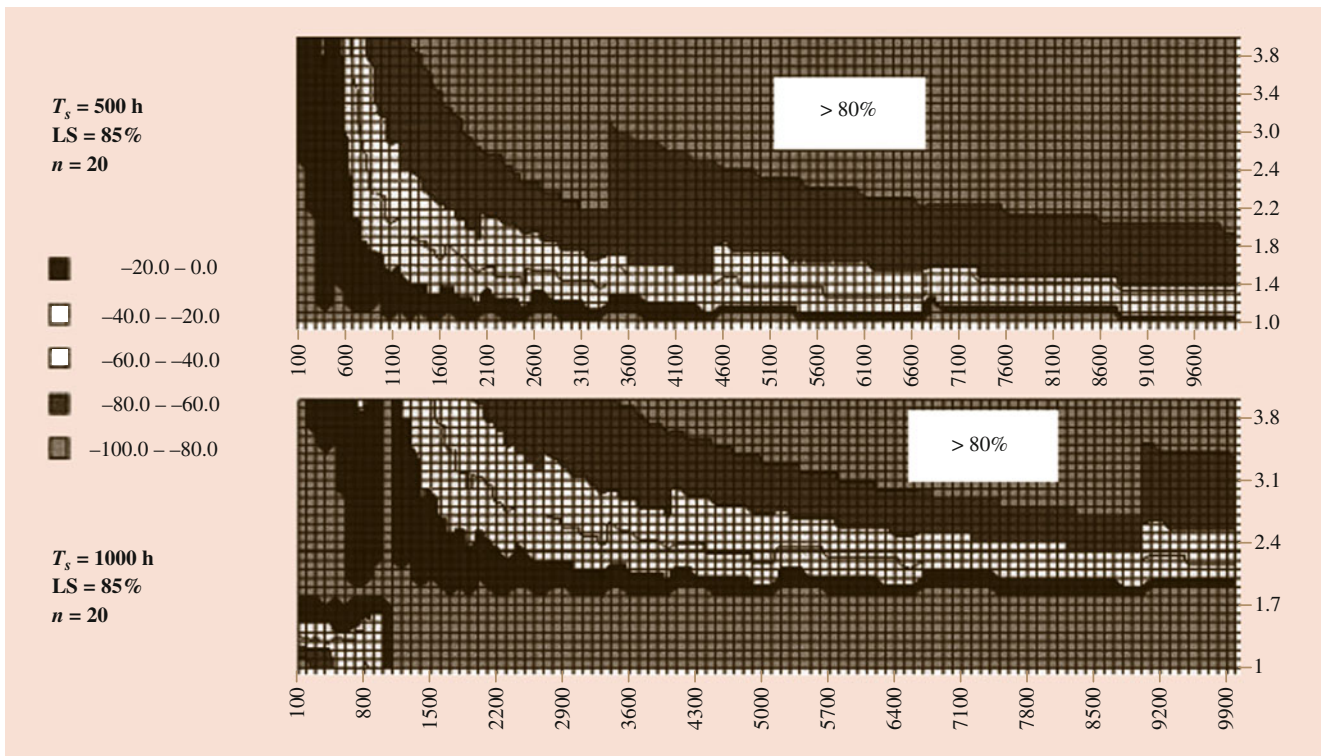


Fig. 15.24 Number of spare parts saved

15.9 A Differentiated Approach for Spare Parts: Standard Versus Custom-Made Components

There are many situations in which it is necessary to manage different components, commercial components and specific components. The former identify the so-called “standard”

materials, which can be found directly on the common spare parts market, while the construction codes are all those custom-made components. Each macrofamily requires differentiated management depending on different factors, such as the differential cost, which often characterizes the construction materials from the commercial, rather than the supply lead times, usually longer for the former. The case study refers to a large-scale plant engineering sector with a spare

Part	Description	PREV	TAB	SS	Stock	LS des	LS stock	T.A.	LT
CM01	CUSCINETTI A COPPIA	420	RD	10	10	0,75	0,00	F40	28
CM02	ROTELLA MCRYD17-3	600	RD	20	127	0,75	1,00	F	28
CM03	STELLA ELAST. 105x26	90	RD	3	92	0,75	1,00	F	28
CM04	GUARNIZIONE ORM 055-10	30000	RD	500	7995	0,75	1,00	F40	28
CM05	CARTUCCIA FIOA 360	240	RD	10	125	0,75	1,00	F	60
CS01	NASTRO	180	RD	5	90	0,75	1,00	F	28
CS02	GUIDA TAPPI	160	RD	10	43	0,75	1,00	F	50
CS03	GUIDA	16	RB	1	18	0,75	1,00	F	28
CS04	STELLA	9	R3	1	1	0,75	0,02	F	28
CS05	TESTA TRANCIANTE D=6	800	RD	30	30	0,75	0,00	F40	90

Fig. 15.25 Reference sample of the codes (CM ## commercial code, CS ## constructive code)

parts park consisting of these two macrofamilies; in order to perform detailed and consistent analyzes, a representative sample of the entire spare parts park was extracted. The sample contains different types of codes (5 commercial and 5 constructive) with different characteristics:

- Codes subject to wear – they must be replaced periodically with new ones; therefore, they present a consumption history characterized by significant movement. These are therefore interesting codes to be analyzed in terms of forecasting.
- Planned codes (Fig. 15.25).

The following information is also shown in the table:

- Current forecast value (PREV) set by the spare parts dealer
- Current safety stock value (SS) set by the spare parts dealer
- Repetitiveness (TAB) set by the spare parts dealer
- Current stock value (Giac)
- Service level (LS Des) imposed by company policy
- Calculated service level (LS Giac) based on the current stock value
- Type of code (T.A.)
- Lead time (LT) set to system

The following data was retrieved for each code:

- Average consumption in the past years (from 01/2013 to 06/2018) [units/month]
- Monthly stock trend (from 01/2013 to 06/2018)
- Total number of applications, calculated on the basis of the number of machines in the park installed, globally, in which the code is mounted, and the number of components present in each equipment
- Average consumption calculated as the reciprocal of the component's running time (Fig. 15.26)

All forecasting methods, except for the Poisson method and the binomial method, use only the consumption history as the database and provide the sales trend; on the contrary, the Poisson method and the binomial method provide a sales forecast value for a given time horizon. The results obtained from the application of the models to the components of the commercial (CM) and construction (CS) categories are critically compared below. The results relating to the codes CM01 and CS01 are reported, to which all the models excluded the Poisson method and the binomial method, and CM02 and CS03, to which only the Poisson method and the binomial method are applied. In the following is the historical consumption and monthly trend of stock patterns five years and a half long [units] for codes CM01 (a coupled bearing), CS01 (a belt), CM02 (a roller), and CS03 (a guide) (Figs. 15.27, 15.28, 15.29, and 15.30):

15.9.1 Performance Analysis of Statistical Models

The performance of statistical methods for the analysis of time series with exponential smoothing parameters depends on how the same smoothing parameters are estimated: Significant improvements in the performance of the forecasting methods can be obtained by further analyzing the historical series of consumption relating to the various codes making up the selected reference sample. The following values were used for the applications discussed:

- For the SES and DES models, the optimal parameters are calculated using an ARIMA model.
- For the Winter models (MW and AW), smoothing parameters are used for the various components equal to 0.2, as recommended by the literature.

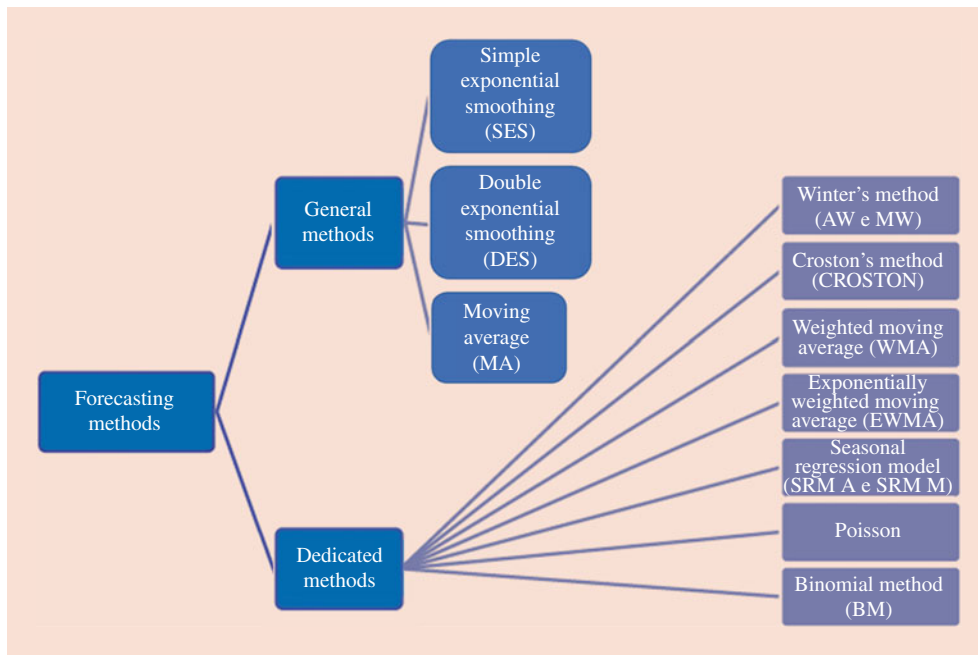


Fig. 15.26 Forecasting methods applied to the reference sample

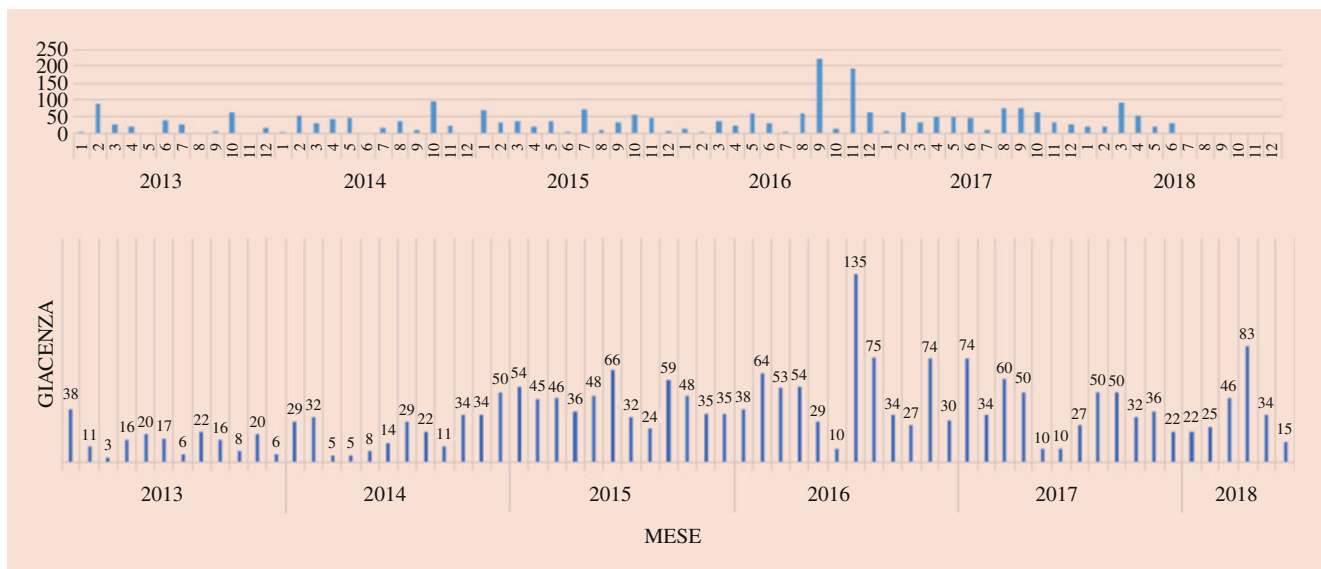


Fig. 15.27 Historical consumption and monthly trend of stock patterns for code CM01

- For the WMA and EWMA models, it is advisable to carry out the analysis of the optimal value of the parameter N , number of periods considered in the time window.
- For the Croston model, it is advisable to carry out the analysis of the smoothing coefficient α .

The proposed Key Performance Indicator for a quantitative evaluation of the predictive capacity of the models is a standardized MAD:

$$KPI = \frac{MAD}{A} = \frac{\sum_{i=1}^n |\varepsilon_{ri} - \varepsilon_{\hat{r}i}|}{\frac{n}{\sum_{i=1}^n \varepsilon_{ri}}}$$

ε_{ri} = real historical consumption data for the i -th period
 $\varepsilon_{\hat{r}i}$ = forecast for the i -th period
 n = number of periods (months)

The indicator was calculated for each code in the sample codes and for all the models considered in order to draw up

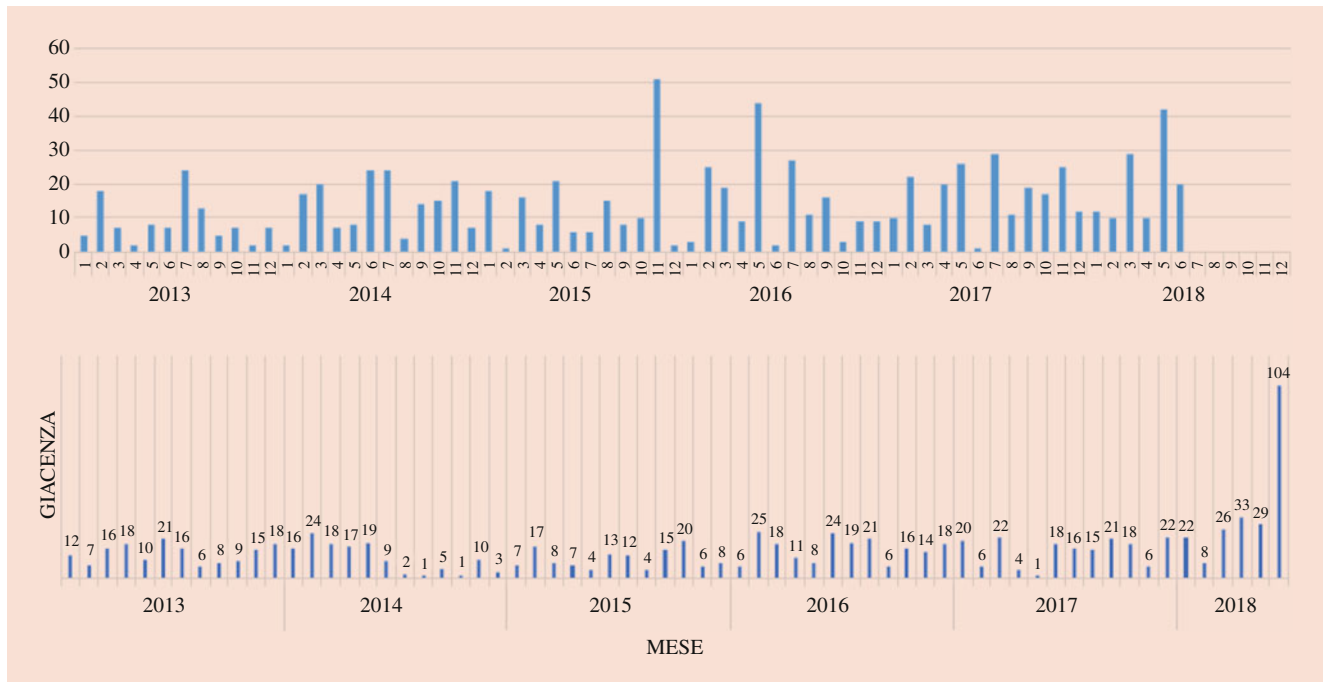


Fig. 15.28 Historical consumption and monthly trend of stock patterns for code CS01

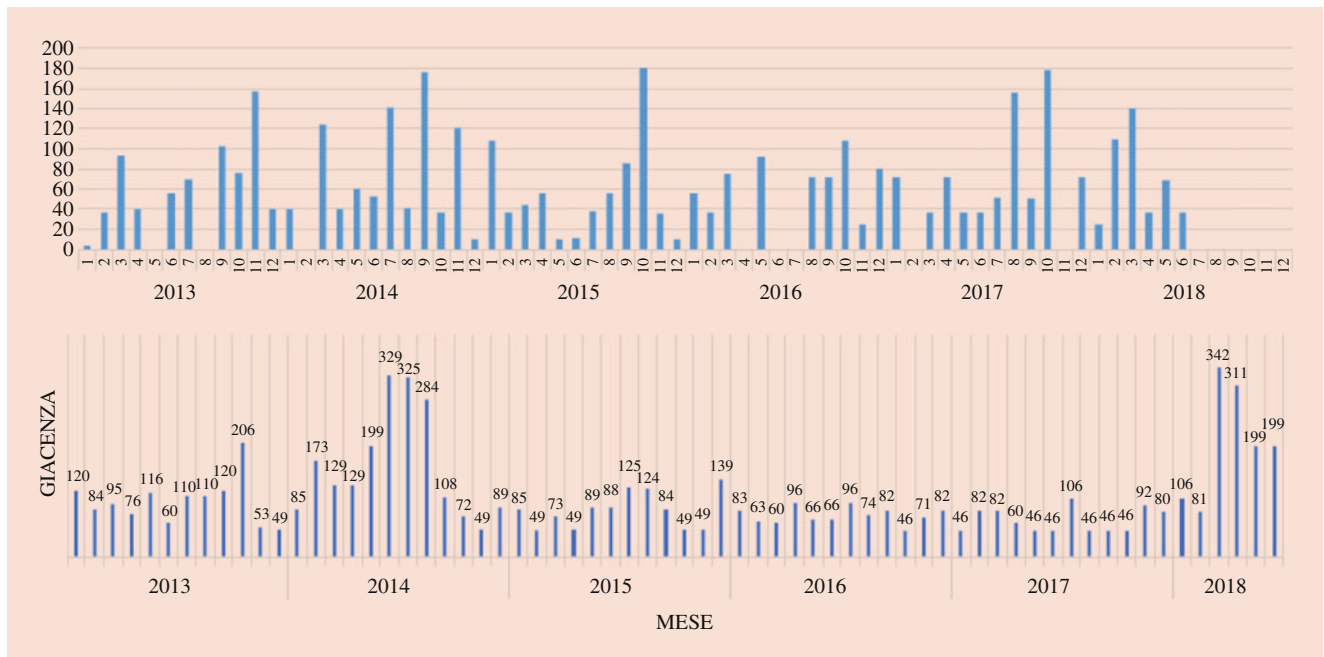


Fig. 15.29 Historical consumption and monthly trend of stock patterns for code CM02

a performance classification; the best performances correspond to MAD/A values much lower than the unit, while the worst ones are characterized by values higher than the unit (Fig. 15.31).

The average value of MAD/A is 0.78, which means that on average the forecasting methods used have performed

well; moreover, best values are obtained for CS03 and CS04 codes having a lumpy demand. The following table shows the ranking of the performance of the different methods for all the codes analyzed (Fig. 15.32).

It is therefore possible to construct a ranking based on the performances obtained by the different methods for the

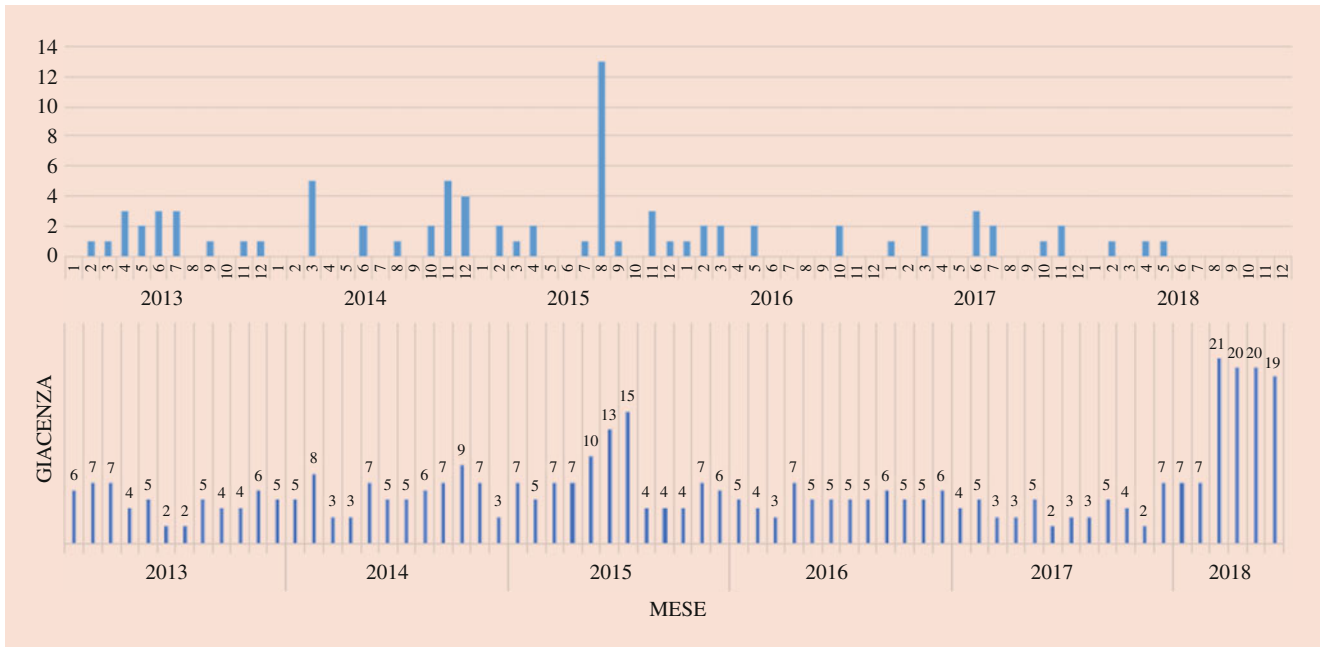


Fig. 15.30 Historical consumption and monthly trend of stock patterns for code CS03

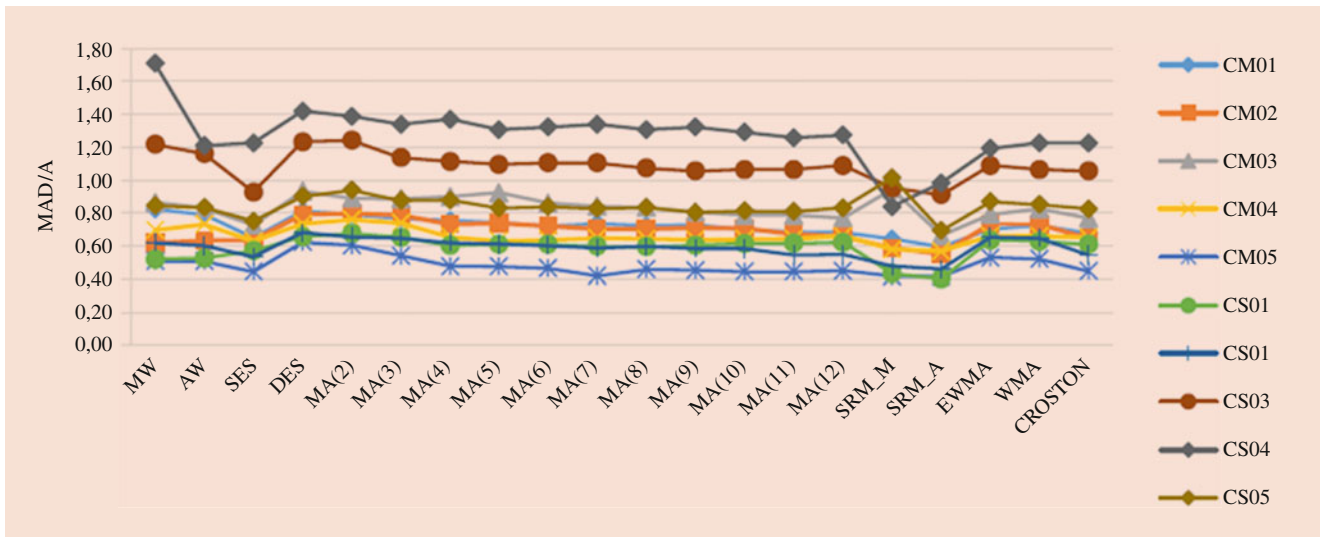


Fig. 15.31 Rend of the MAD/A indicator by code for different methods

various codes, associating a score to each method, corresponding to its position. For instance, Croston method ranks in fourth position for CM01, in sixth position for CM02, etc.; to this model, 4 points for CM01, 6 points for CM02, etc., are assigned, for a total amount of 62 points. On the basis of this allocation criterion, the lower the score, the better the overall performance of the model. The Additive Seasonal Regression Model (SRM_A) appears to be the most applicable forecasting method to the spare parts fleet.

15.9.2 Performance Analysis of Poisson and Binomial Models

The same methodology of investigation was adopted for the Poisson method and the binomial method, as they can be easily compared; since they both provide a prediction concerning the number of pieces that will be consumed during a certain time horizon, set a certain level of service (LS). The following tables show the forecast values provided by

Posizione	CODICI COMMERCIALI					CODICI COSTRUTTIVI				
	CM01	CM02	CM03	CM04	CM05	CS01	CS02	CS03	CS04	CS05
1	SRM_A	SRM_A	SRM_A	SRM_A	SRM_A	SRM_A	SRM_A	SRM_A	SRM_M	SRM_A
2	SRM_M	SRM_M	SES	SRM_M	SRM_M	SRM_M	SRM_M	SES	SRM_A	SES
3	SES	MW	CROSTON	SES	MA(7)	MW	SES	SRM_M	EWMA	MA(9)
4	CROSTON	AW	MA(12)	MA(5)	SES	AW	CROSTON	CROSTON	AW	MA(11)
5	MA(12)	SES	MA(11)	MA(9)	MA(10)	SES	MA(11)	MA(9)	SES	MA(10)
6	MA(11)	CROSTON	MA(10)	MA(6)	MA(11)	MA(8)	MA(12)	MA(11)	CROSTON	CROSTON
7	MA(10)	MA(12)	EWMA	MA(10)	CROSTON	MA(7)	MA(10)	MA(10)	WMA	MA(7)
8	EWMA	MA(11)	MA(9)	MA(11)	MA(12)	MA(4)	MA(9)	WMA	MA(11)	MA(5)
9	MA(6)	MA(8)	WMA	MA(8)	MA(9)	MA(9)	MA(7)	MA(8)	MA(12)	MA(12)
10	WMA	MA(7)	AW	MA(7)	MA(8)	MA(6)	MA(8)	MA(12)	MA(10)	AW
11	MA(8)	MA(10)	MA(8)	CROSTON	MA(6)	CROSTON	AW	EWMA	MA(8)	MA(8)
12	MA(9)	MA(9)	MA(7)	MA(4)	MA(5)	MA(5)	MA(6)	MA(5)	MA(5)	MA(6)
13	MA(7)	MA(6)	MA(6)	WMA	MA(4)	MA(10)	MA(5)	MA(7)	MA(9)	MW
14	MA(5)	WMA	MW	EWMA	MW	MA(11)	MW	MA(6)	MA(6)	WMA
15	MA(4)	MA(4)	MA(3)	MA(12)	AW	MA(12)	MA(4)	MA(4)	MA(7)	EWMA
16	MA(3)	EWMA	MA(2)	MW	WMA	WMA	MA(3)	MA(3)	MA(3)	MA(3)
17	AW	MA(5)	MA(4)	AW	EWMA	EWMA	WMA	AW	MA(4)	MA(4)
18	MA(2)	MA(3)	MA(5)	DES	MA(3)	MA(3)	EWMA	MW	MA(2)	DES
19	DES	DES	DES	MA(3)	MA(2)	DES	MA(2)	DES	DES	MA(2)
20	MW	MA(2)	SRM_M	MA(2)	DES	MA(2)	DES	MA(2)	MW	SRM_M

Fig. 15.32 Ranking of the performance of the adopted forecasting methods

	Consumption history						
	T1 = 12 months						
	PREV	TAB	SS	N° of applications	n° parts LS ≥ 75%	n° parts LS ≥ 85%	n° parts LS ≥ 95%
CM01	420	RD	10	9866	483	491	504
CM02	600	RD	20	1140	740	750	757
CM03	90	RD	3	572	100	103	109
CM04	30000	RD	500	1189	26014	26072	26170
CM05	240	RD	10	771	271	277	287
CS01	180	RD	5	503	178	183	191
CS02	160	RD	10	61	172	177	185
CS03	16	RB	1	58	17	19	21
CS04	9	R3	1	47	9	10	12
CS05	800	RD	30	1202	975	987	1006

Fig. 15.33 Forecast values based on the consumption history with the Poisson method

the Poisson method and the binomial method, as regards the consumption history (Figs. 15.33 and 15.34).

By cross-checking the forecast values obtained by the Poisson model, with those generated by the binomial method, the fact immediately emerges that the former, with the same time horizon and service level adopted, are greater than the

latter. This is in accordance with the technical literature, from which it is derived that the binomial method was precisely introduced as a development of the application of the Poisson formula with which often inaccurate and tending to overestimate forecasts are recorded. It can therefore be stated that based on the information and the methods with which

	PREV	TAB	SS	N° of applications	consumption history		
					T1 = 12 months		
					n° parts LS ≥ 75%	n° parts LS ≥ 85%	n° parts LS ≥ 95%
CM01	420	RD	10	9866	471	479	492
CM02	600	RD	20	1140	546	552	563
CM03	90	RD	3	572	92	95	100
CM04	30000	RD	500	1189	25629	25635	25645
CM05	240	RD	10	771	229	234	241
CS01	180	RD	5	503	150	154	160
CS02	160	RD	10	61	155	156	168
CS03	16	RB	1	58	15	16	19
CS04	9	R3	1	47	8	9	11
CS05	800	RD	30	1202	670	677	687

Fig. 15.34 Forecast values based on the consumption history with the binomial method

Codice	MAD	A	MAD/A
CM01	131,33	256,67	0,512
CM02	114,33	391,00	0,292
CM03	15,33	60,00	0,256
CM04	3503,67	17724,00	0,198
CM05	36,00	126,33	0,285
CS01	18,67	107,67	0,173
CS02	29,67	81,00	0,366
CS03	2,67	4,67	0,571
CS04	3,33	3,33	1,000
CS05	155,33	419,67	0,370

Fig. 15.35 Values of MAD/A related to the binomial method

the forecast analyses were conducted, the binomial method is more efficient than the Poisson model. Also, for this comparison the same KPI used for the comparison between the statistical models was used.

From the observation of the obtained MAD/A values, it can immediately be deduced that, CS04 code apart, the binomial method has excellent performances. The average value of MAD/A is 0.40, which means that the binomial method is the best performing for the spare fleet (Fig. 15.35).

15.9.3 Operational Management of Spare Parts

In addition to identifying the best method for forecasting the quantity of parts for each spare part that the market

will require, the company must also be able to operatively manage its own codes, with the aim of achieving a correct optimization policy. Therefore, the company will have to simultaneously determine the spare parts that will be consumed in the coming **months**, using one of the forecasting models studied, and define what will actually be the correct quantity of components to keep in stock over a given time horizon. The objective of the criterion of minimum global cost is to determine the optimal number of spare parts to keep in stock corresponding to the best trade-off between the overall cost related to their management.

$$\min C_{tot}(N) = C_1 + C_2$$

N = optimal number of spare parts of the code to keep in stock

$C_1 = R \cdot \varphi \cdot [N \cdot pd,T,0 + (N - 1) \cdot pd,T,1 + (N - 2) \cdot pd,T,2 + \dots + pd,T,N - 1]$ storage cost

T = duration of the procurement cycle [unit of time]; it is the sum of the time of issue of the purchase order and the procurement time

d = average code consumption per unit of time [units/unit of time]; it is the average value based on the consumption history

$P_{d,T,x}$ = probability that x requests of the same spare will occur in the interval T

R = purchase cost of the code [€]

φ = distribution overhead [%]; it is the percentage rate expressing the cost of stock

$R \cdot \varphi$ = specific annual cost of storage [€/year]

$C_2 = C_m \cdot d \cdot P$ stock-out cost

C_m = unit cost of stock-out [€/units]

P = probability of higher demand for N spare parts in the interval T

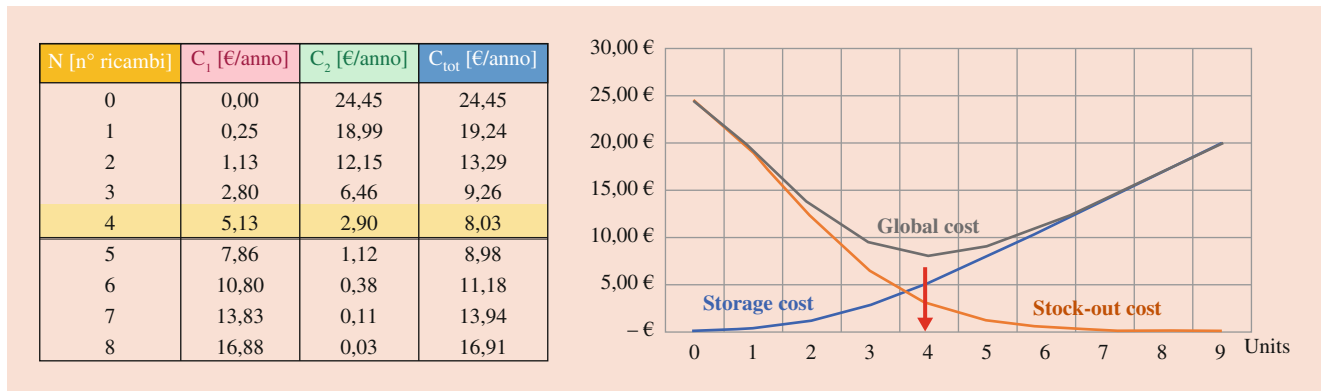


Fig. 15.36 Optimal units in stock and trade-off for the CS04 code ($d = 0.62$ units/month, $T = 4033$ months, $R = 9,59$ €, $\rho = 0.2$)

$$P = P_{d,T,N+1} + P_{d,T,N+2} + P_{d,T,N+3} + \dots$$

$$= 1 - (P_{d,T,0} + P_{d,T,1} + P_{d,T,2} + \dots + P_{d,T,N})$$

The distribution overhead φ depends on the fixed and variable costs related to the management of the materials in stock and the number of movements in a year. The following cost items are considered as follows (annual costs):

- Overhead costs on the “warehouse cost center”
- Cost of the staff involved
- Cost of utilities (electricity, methane gas...)
- Cost for warehouse maintenance
- Cost of material handling
- Depreciation
- Other costs

The total amount of these costs is spread over the number of units handled in a year for each code; this means that the costs of the spare parts in stock, with zero rotation index, are then divided between the materials with nonzero rotation index.

The determination of the costs of stock-out is different for a manufacturing company with respect to a user: The unavailability of a spare part for a customer can have different effects on company costs, depending on different factors; among them may be the importance of this spare code for the requesting customer, which fundamentally depends on the possibility that the unavailability of the component entails the nonproduction of the customer. This situation is reflected in the issue of an urgent request by the customer. In other cases where production is not stopped, for example, for material required to replenish the customer’s spare parts warehouse, the customer’s request follows an ordinary procedure: In these cases, there is no economic damage for the manufacturing company (Fig. 15.36).

$$C_m = \alpha \cdot R + V$$

α = percentage corresponding to the increase in the purchase cost that the producer must incur for his emergency supply [%]; in this case study, $\alpha = 0.1$ and $\alpha = 0.05$ for construction and commercial components, respectively
 V = sale price of the code to the customer [€]

References

1. Ghobbar, A.A., Friend, C.H.: Sources of intermittent demand for aircraft spare parts within airline operations. *J. Air Transp. Manag.* **8**, 221–231 (2002)
2. Syntetos, A.: Forecasting of intermittent demand. Ph.D Thesis, Buckinghamshire Business Sch, Brunel University (2001)
3. Williams, T.M.: Stock control with sporadic and slow-moving demand. *J. Oper. Res. Soc.* **35**(10), 939–948 (1984)
4. Campbell, H.S.: The Relationship of Resource Demands to Air Base Operations. RAND, Santa Monica (1963)
5. Clarke, M.D.: Irregular airline operations: a review of the state of the practice in airline operations control centres. *J. Air Transp. Manag.* **4**(2), 67–76 (1998)
6. Friend, C.H.: Aircraft Maintenance Management. Longman, Scientific & Technical, Harlow (1992)
7. Willemain, T.R., Smart, C.N., Schwarz, H.F.: A new approach to forecasting intermittent demand for service parts inventories. *Int. J. Forecast.* **20**, 375–387 (2004)
8. Lau, H.S., Wang, M.C.: Estimating the lead-time demand distribution when the daily demand is non-normal and autocorrelated. *Eur. J. Oper. Res.* **29**, 60–69 (1987)
9. Tyworth, J.E., O’Neill, L.: Robustness of the normal approximation of lead-time demand in a distribution setting. *Nav. Res. Logist.* **44**, 165–186 (1997)
10. Ward, J.B.: Determining reorder points when demand is lumpy. *Manag. Sci.* **24**, 623–632 (1978)
11. Mitchell, C.R., Rappold, R.A., Faulkner, W.B.: An analysis of Air Force EOQ data with an application to reorder point calculation. *Manag. Sci.* **29**, 440–446 (1983)
12. Van Ness, P.D., Stevenson, W.J.: Reorder-point models with discrete probability distributions. *Decis. Sci.* **14**, 363–369 (1983)
13. Schultz, C.R.: Forecasting and inventory control for sporadic demand under periodic review. *J. Oper. Res. Soc.* **37**, 303–308 (1987)
14. Watson, R.B.: The effects of demand-forecast fluctuations on customer service and inventory cost when demand is lumpy. *J. Oper. Res. Soc.* **38**, 75–82 (1987)

15. Dunsmuir, W.T.M., Snyder, R.D.: Control of inventories with intermittent demand. *Eur. J. Oper. Res.* **40**, 16–21 (1989)
16. Petrovic, D., Petrovic, R.: SPARTA II: further development in an expert system for advising on stocks of spare parts. *Int. J. Prod. Econ.* **24**, 291–300 (1992)
17. Bier, I.J.: Boeing Commercial Airplane Group Spares Department: Simulation of Spare Parts Operations. ORSA/TIMS Joint National Meeting, Detroit (1984)
18. Sani, B., Kingsman, B.G.: Selecting the best periodic inventory control and demand forecasting methods for low demand items. *J. Oper. Res. Soc.* **48**, 700–713 (1997)
19. Croston, J.D.: Forecasting and stock control for intermittent demands. *Oper. Res. Q.* **23**, 289–303 (1972)
20. Segerstedt, A.: Inventory control with variation in lead times, especially when demand is intermittent. *Int. J. Prod. Econ.* **35**, 365–372 (1994)
21. Johnston, F.R., Boylan, J.E.: Forecasting for items with intermittent demand. *J. Oper. Res. Soc.* **47**, 113–121 (1996)
22. Bookbinder, J.H., Lordahl, A.E.: Estimation of inventory re-order levels using the bootstrap statistical procedure. *IIE Trans.* **21**, 302–312 (1989)
23. Wang, M., Rao, S.S.: Estimating reorder points and other management science applications by bootstrap procedure. *Eur. J. Oper. Res.* **56**, 332–342 (1992)
24. Kim, Y.B., Haddock, J., Willemain, T.R.: The binary bootstrap: inference with autocorrelated binary data. *Commun. Stat. Simul. Comput.* **22**, 205–216 (1993)
25. Park, D., Willemain, T.R.: The threshold bootstrap and threshold jackknife. *Comput. Stat. Data Anal.* **31**, 187–202 (1999)
26. Ghobbar, A.A., Friend, C.H.: Evaluation of forecasting methods for intermittent parts demand in the field of aviation: a predictive model. *Comput. Oper. Res.* **30**(14), 2097–2114 (2003)
27. Bartezzaghi, E., Verganti, R., Zotteri, G.: A simulation framework for forecasting uncertain lumpy demand. *Int. J. Prod. Econ.* **59**, 499–510 (1999)
28. Brown, R.G.: *Statistical Forecasting for Inventory Control*. McGraw-Hill, New York (1959)
29. Alstrom, P., Madsen, P.: Tracking signals in inventory control systems. A simulation study. *Int. J. Prod. Econ.* **45**(1–3), 293–302 (1996)
30. Coughlin R.: Optimization of spares in maintenance scenario. In: *Proceedings of Reliability and Maintainability Symposium*, San Francisco (1984)
31. Metaweh A.: A cost reliability model with spares in electric power system. In: *Proceedings of reliability and maintainability symposium*, Philadelphia (1997)
32. Pareschi, A., Persona, A., Regattieri, A., Ferrari, E.: TPM: a total approach for industrial maintenance question. In: *Proceedings of 7th International Conference on reliability and quality in design – ISSAT 2001*, 8–10 August – Washington D.C. pp. 262–268
33. Pareschi, A., Persona, A., Regattieri, A.: Methodology for spare parts level optimization in maintenance systems. In: *Proceedings ISSAT 2000*, 9–11 August, Orlando
34. Persona, A., Regattieri, A., Catena, M.: Low turnaround spare parts level optimization. In: *Proceedings of 9th International Conference on Reliability and Quality in Design – ISSAT 2003*, August – Honolulu (Hawaii), pp. 273–277
35. Pareschi, A., Persona, A., Regattieri, A., Ferrari, E.: TPM: situation and procedure for a soft introduction in Italian factories. *TQM Mag.* **14**(6), 350–358
36. Hjorth, U.: A reliability distribution with increasing, decreasing, constant and bathtub-shaped failure rates. *Technometrics*, 22–99 (1980)
37. Mudholkar, G.S., Srivastava, D.K.: Exponentiated Weibull family for analyzing bathtub failure-rate data. *IEEE Trans. Reliab.* **44**, 388–391 (1993)
38. Chen, Z.: A new two-parameter lifetime distribution with bathtub shape or increasing failure rate function. *Stat. Prob. Lett.* **49**, 155–161 (2000)
39. Xie, M., Tang, Y., Goh, T.N.: A modified Weibull extension with bathtub-shaped failure rate function. *Reliab. Eng. Syst. Saf.*, 279–285 (2002)



Emilio Ferrari received his MSc in Mechanical engineering at the University of Bologna in 1988. He worked in the industrial sector of packaging. He is Full Professor in Industrial Logistics, Past Deputy Rector of Bologna University, and President of CINECA, Consortium of Universities and Government for services and HPC. He works now on the design and optimization of industrial production systems.



Arrigo Pareschi graduated in Nuclear Engineering at Bologna University in 1969. He was full professor of “Industrial Plants,” Dean of Engineering Faculty (1995–2001), and President of Scientific Research Observatory in the same University. He is author/coauthor of 105 scientific papers, concerning industrial production systems, logistics, and technological innovation and author of four books of interest for university students and engineers.



Alberto Regattieri is a full professor in Logistics and Head of the Management Engineering Master Course at the University of Bologna (Italy). His current research interests include the optimal design of manufacturing systems, innovative approaches (based on Industry 4.0 key technologies) to design and manage handling and stocking systems, to control and manage maintenance of industrial plants. He is/was responsible of several research projects in cooperation with – and funded by – European Commission, private and public companies, universities, and international research centers regarding logistics and manufacturing processes. He published more than 170 scientific papers.



Alessandro Persona is full professor in Logistics at the University of Padua. His current research interests deal with Production, Operations management, Operations research, Supply chain and Flexibility. His Production research includes elements of Industrial engineering, Risk analysis, Relevant cost, Optimal maintenance and Process. His research in the fields of Spare part and Stock-taking overlaps with other disciplines such as Information technology and Order.



D-Efficient Mixed-Level Foldover Designs for Screening Experiments

16

Nam-Ky Nguyen , Ron S. Kenett , Tung-Dinh Pham , and Mai Phuong Vuong

Contents

16.1	Introduction	305
16.2	A CEA for Constructing D-efficient ADSDs	306
16.3	An ESA for Constructing D-efficient MLFODs	307
16.4	Results and Discussion	308
16.5	An Industrial Case Study	308
16.6	Conclusion	311
	Appendix: Calculating (d_1, d_2) Values of an MLFOD	311
	References	312

Abstract

Definitive screening design (DSD) is a new class of three-level screening designs proposed by Jones and Nachtsheim [3] which only requires $2m + 1$ runs for experiments with m three-level quantitative factors. The design matrices for DSDs are of the form $(\mathbf{C}', -\mathbf{C}', \mathbf{0})'$ where \mathbf{C} is a $(0, \pm 1)$ submatrix with zero diagonal and $\mathbf{0}$ is a column vector of 0's. This paper reviews recent development on D-efficient mixed-level foldover designs for screening experiments. It then discusses a fast coordinate-exchange algorithm for constructing D-efficient DSD-augmented designs (ADSDs). This algorithm provides a new class

of conference matrix-based mixed-level foldover designs (MLFODs) for screening experiments as introduced by Jones and Nachtsheim [4]. In addition, the paper also provides an alternative class of D-efficient MLFODs and an exhaustive algorithm for constructing the new designs. A case study comparing two candidate MLFODs for a large mixed-level screening experiment with 17 factors used is used to demonstrate the properties of the new designs.

Keywords

Conference matrix · Coordinate-exchange algorithm · Foldover design · Hadamard matrix · Plackett-Burman design

16.1 Introduction

Screening experiments are designed to sort a typically long list of factors that can potentially affect the response variables of a product or process. The sorting highlights active factors. This experimentation strategy is widely applied in science and engineering. Another approach, pioneered by Genichi Taguchi, is to follow an experimental path of system design, parameter design, and tolerance design [6]. In this paper we consider the screening-optimizing continuum with the objective of improving the knowledge acquisition effort by increasing its quality and reducing the required effort. Most screening experiments in engineering and science involve both two-level and three-level factors. Yet, the most popular screening designs are two-level designs such as resolution III and IV fractional factorial designs (FFDs). Jones and Nachtsheim [3] pointed out the following disadvantages of using two-level FFDs to study quantitative factors:

- (i) Quadratic effects are not estimable if they are included in the model;

N.-K. Nguyen (✉)
Vietnam Institute for Advanced Study in Mathematics, Hanoi, Vietnam
e-mail: nknam@viasm.edu.vn

R. S. Kenett
KPA Ltd., Samuel Neaman Institute, Technion, Israel
e-mail: ron@kpa-group.com

T.-D. Pham
VNU University of Science, Hanoi, Vietnam
e-mail: tungpd@vnu.edu.vn

M. P. Vuong
Hanoi University of Science & Technology, Hanoi, Vietnam
e-mail: phuong.vuongmai@hust.edu.vn

- (ii) Main effects are not completely orthogonal to two-factor interactions as in the case of resolution III FFDs;
- (iii) Certain two-factor interactions are fully aliased with one another as in the case of resolution IV FFDs;

A new class of three-level screening designs called DSDs introduced by Jones and Nachtshiem [3] eliminates these shortcomings. In addition, all quadratic effects of DSDs are orthogonal to main effects and not fully aliased with two-factor interactions. The design matrix for a DSD can be written as:

$$\begin{pmatrix} \mathbf{C} \\ -\mathbf{C} \\ \mathbf{0}' \end{pmatrix}, \tag{16.1}$$

where \mathbf{C} is an $m \times m$ $(0, \pm 1)$ submatrix with zero diagonal and $\mathbf{0}$ is a column vector of 0's. Xiao et al. [14] pointed out that if we use a *conference* matrix of order m for \mathbf{C} , i.e., if $\mathbf{C}'\mathbf{C} = (m - 1) \mathbf{I}_{m \times m}$, then the DSD is also orthogonal for main effects, i.e., all main effects are orthogonal to one another. For even $m \leq 50$ except for $m = 22$ and $m = 34$, the \mathbf{C} matrices which are also conference matrices are given by Xiao et al. [14] and Nguyen and Stylianou [11]. All the \mathbf{C} matrices we use in this paper are conference matrices with the exception of the one of order 22. Figure 16.1 shows the \mathbf{C} matrices of order $m = 4, 6, 8,$ and 10 generated by the cyclic generators given by Nguyen and Stylianou [11].

The limitation of a DSD is that all factors should be quantitative. Jones and Nachtshiem [4] (hereafter abbreviated as JN) introduced two types of conference matrix-based mixed-level screening designs. They called the more D-efficient, more economic one, DSD-augmented designs (ADSDs). ADSDs are in fact belonging to a class of mixed-level foldover designs (MLFODs), and, as such, they retain two advantages of the original DSD, namely, (i) all quadratic effects are orthogonal to main effects and (ii) all main effects are orthogonal to two-factor interactions. The latter feature is extremely useful when the experimenter wishes to include a

specific two-factor interaction in the model. The limitation of this class of designs is that it has a high correlation among the quadratic effects. JN showed that this correlation was $\frac{1}{2} - \frac{2}{n-4}$ where n is the number of runs.

Nguyen et al. [12] introduced a new class of Hadamard matrix-based mixed-level foldover designs (MLFODs) and an algorithm which produces these MLFODs. These MLFODs were constructed by converting some two-level columns of a Hadamard matrix to three-level ones (see [2] for the information on Hadamard matrices and their use in design construction). Like the two-level foldover designs (FODs), each new MLFOD was constructed by a half fraction and its foldover. These Hadamard matrix-based MLFODs require fewer runs and compare favorably with the conference matrix-based MLFODs of [4] in terms of the D-efficiencies and r_{max} , the maximum of the absolute values of the correlation coefficients among the columns of the model matrix. Like the ADSDs, these MLFODs are also *definitive* in the sense that the estimates of all main effects are unbiased with respect to any active second-order effects. The limitation of this class of Hadamard matrix-based MLFODs is that it is not very efficient when the number of three-level factors is greater than the number of two-level ones.

The design matrix of a conference matrix-based MLFOD for m_3 three-level factors and m_2 two-level factors has a foldover structure and can be written as:

$$\begin{pmatrix} \mathbf{D} \\ -\mathbf{D} \end{pmatrix}, \tag{16.2}$$

where the submatrix $\mathbf{D}_{m^* \times (m_3+m_2)}$ can be constructed from a matrix of order m ($m_3 + m_2 \leq m \leq m^*$). In the following sections, we describe (i) a fast coordinate-exchange algorithm (CEA) [8] for constructing ADSDs which are conference matrix based and (ii) an exhaustive search algorithm or ESA for constructing an alternative class of MLFODs. Both algorithms attempt to transform a base matrix to the submatrix \mathbf{D} in (16.2) from which a D-efficient MLFOD can be obtained. For ADSDs, the base matrix is a conference matrix. For the new MLFODs, the base matrix is a two-level orthogonal matrix such as Hadamard matrix or a Plackett-Burman design [13].

m=4	m=6	m=8	m=10
0+++	0+++++	0+++++++	0+---++++
+0--	+0---+	+0+++--	+0---+----
++0-	+--0++	+--0+++--	--0+----++
+--+0	++-0+-	++-0++++	---0+++++
	+++--0-	+++--0++-	+++--0++++-
	++++-0	++++-0++	++++-0+++-
		++++--0+	++++--0+++
		++++---0	++++---0++
			++++---0+-
			++++---0--
			++++----0

Fig. 16.1 Conference \mathbf{C} matrices of order m (+ denotes +1 and - denotes -1)

16.2 A CEA for Constructing D-efficient ADSDs

The Appendix shows that the first-order and second-order D-efficiencies d_1 and d_2 of an ADSD are functions of $|\mathbf{B}|$ ($= |\mathbf{D}'\mathbf{D}|$) where $|\mathbf{B}|$ is the determinant of matrix \mathbf{B} . Our algorithm minimizes the sum of squares of the off-diagonal elements of \mathbf{B} as an indirect attempt to maximize $|\mathbf{B}|$

(see, e.g., [9]). The steps of our CEA for obtaining a submatrix **D** in (2) from which a D-efficient ADSD for m_3 three-level factors and m_2 two-level factors are:

1. Form a starting design $\mathbf{D}_{(m+1) \times (m_3+m_2)} = (d_{ij})$ by first picking $m_3 + m_2$ columns from a conference matrix of order m at random and add an extra zero row to the bottom of these columns. Mark the positions of the $2m_2$ zero entries in the last m_2 columns of **D**, and replace these 0's by ± 1 in a random manner. Calculate the vector \mathbf{J}_i ($i = 1, \dots, m + 1$) of length $m_3m_2 + \binom{m_2}{2}$ for each row of **D** where \mathbf{J}_i is defined as $(d_{i1}d_{i(m_3+1)}, \dots, d_{im_3}d_{i(m_3+m_2)}, d_{i(m_3+1)}d_{i(m_3+2)}, \dots, d_{i(m_3+m_2-1)}d_{i(m_3+m_2)})$. Let $\mathbf{J} = \sum_{i=1}^{m+1} \mathbf{J}_i$ and f equal the sum of squares of the elements of **J**.
2. Among the $2m_2$ marked positions in Step 1, search for a position such that the sign switch in this position results in the biggest reduction in f . If the search is successful, update f , **J** and **D**. Repeat this step until f equals the length of **J** (i.e., all elements of **J** equal ± 1) or this value cannot be reduced further by any sign switch.

Remarks

- (i) The above steps correspond to one “try” of the CEA and each try produces a matrix **D**. Among a large number of tries whose f value reaches its lower bound, i.e., f equals **J**'s length (or f cannot be reduced further), the one with the largest value of $|\mathbf{B}|$ is selected.
- (ii) The purpose of picking at random $m_3 + m_2$ columns from a conference matrix of order m ($m \geq m_3 + m_2$) to form **D** in a random manner is to avoid being trapped in the local optima.
- (iii) When the input matrix in Step 1 is a conference matrix, the first m_3m_2 elements of **J** always take values ± 1 . This is not the case when the input matrix is not a conference matrix such as the one of order 22.
- (iv) Our CEA is less prone to the curse of dimensionality than JN's exhaustive algorithm for ADSD construction which attempts to maximize $|\mathbf{B}|$ from among the 2^{2m_2} arrangements for $2m_2$ entries in **D**.

Figure 16.2 shows the steps of constructing an ADSD for four three-level factors and four two-level factors. Figure 16.2a displays a starting design in Step 1. Figure 16.2b shows that the eight 0's in the last four columns of the design in Fig. 16.2a are being replaced by ± 1 in a random manner. At this point, the vector **J** is (1, 1, 1, 1, 1, -1, 1, 1, 1, 1, -1, -1, -1, 1, 1, -3, 1, -1, 1, -1, 3) and f is 38. Figure 16.2c,d correspond to Step 2. In Fig. 16.2c, the value 1 in the position (1, 7) of the design in Fig. 16.2b is replaced by -1. At this point,

a)	b)	c)	d)
+++++0+	+++++++	+++++--+	+++++--+
---+0+++	---++++	---++++	---++++
++0---+-	++0---+-	++0---+-	++0---+-
+--00++	+--0+++	+--0+++	+--0+++
-+0-++	-+0-++	-+0-++	-+0-++
++-+---0	++-+---+	++-+---+	++-+---+
+0-----	+0-----	+0-----	+0-----
0-++++--	0-++++--	0-++++--	0-++++--
00000000	0000+---	0000+---	0000+---

Fig. 16.2 Steps of constructing an ADSD for four three-level factors and four two-level factors

the vector **J** is (1, 1, -1, 1, 1, -1, -1, 1, 1, 1, -1, -1, -1, -1, 1, -3, -1, -1, -1, -1, 1) and f is reduced to 30. In Fig. 16.2d, the value -1 in the position (2, 5) of the design in Fig. 16.2c is replaced by 1. At this point, the vector **J** is (-1, 1, -1, 1, -1, -1, -1, 1, -1, 1, -1, -1, 1, -1, -1, 1, -1, 1, 1, -1, -1) and f is reduced to 22 which is its lower bound.

16.3 An ESA for Constructing D-efficient MLFODs

While the CEA attempts to convert some three-level columns of a conference matrix into two-level columns, the ESA attempts to convert some two-level columns of a two-level orthogonal matrix into three-level columns. The ESA requires three simple steps:

1. From each base matrix of order m , generate $m - 1$ additional matrices by shifting the columns of this matrix to the left cyclically. From each matrix, use the first $m_3 + m_2$ ($\leq m$) columns to form a starting design $\mathbf{D}_{m \times (m_3+m_2)} = (d_{ij})$.
2. For each matrix obtained from Step 1, generate $\binom{m}{k}$ new matrices by replacing k elements in each of its first m_3 columns by 0's. Here $k = 2, \dots, x$ where x is an integer chosen to be $\lceil \frac{m}{5} \rceil$ where $\lceil \cdot \rceil$ denotes the ceiling function. The replacement is performed so that if $(i_1, j), \dots, (i_k, j)$ are entries in column j being replaced by 0's, and then the entries being replaced by 0's in the next column are $((i_1 + 1) \bmod m, j + 1), ((i_2 + 1) \bmod m, j + 1), \dots, ((i_k + 1) \bmod m, j + 1)$.
3. For each matrix in Step 2, calculate r_{max} , the maximum in terms of absolute value of the correlation coefficients among the columns of the model matrix. Then among the designs with smallest r_{max} , pick the one with the highest d_2 , the D-efficiency for the pure quadratic model (see Eq. (16.8) in the Appendix).

(1)	(2)	(3)	(4)
0++++++	0++++++	0++++++	0++++++
00+++-	+0+++-	+0+++-	+0+++-
+00++-	0-0++-	+0++-	+0++-
+00++-	0-0++-	+0++-	+0++-
+00++-	0-0++-	+0++-	+0++-
+00++-	0-0++-	+0++-	+0++-
+00++-	0-0++-	+0++-	+0++-
+00++-	0-0++-	+0++-	+0++-
+00++-	0-0++-	+0++-	+0++-
+00++-	0-0++-	+0++-	+0++-

Fig. 16.3 Some candidate designs generated in Step 2 for an MLFOD for four three-level factors and four two-level factors

Remarks

- (i) In Step 1, the base matrix is a two-level orthogonal matrix. This is a Hadamard matrix or a Plackett-Burman design if m is divisible by 4 or a conference matrix with the 0's on the diagonal being replaced by 1's if m is not divisible by 4 but is divisible by 2. The base matrix slightly affects the goodness of the resulting design.
- (ii) For small m , say $m \leq 12$, the starting designs in Step 1 can also be constructed by randomly selecting a subset of $m_3 + m_2$ from m columns of the base matrix.
- (iii) For each pair (m_3, m_2) and a given x , the number of zeros in each of the m_3 three-level columns, the number of candidate designs we have to consider is $m \sum_{k=2}^x \binom{m}{k}$.
- (iv) r_{max} in Step 3 is calculated from the vector $\mathbf{J} = \sum_{i=1}^m \mathbf{J}_i$ of length $2\binom{m_3}{2} + m_3m_2$ where \mathbf{J}_i is defined as $(d_{i1}^2 d_{i2}^2, \dots, d_{i(m_3-1)}^2 d_{im_3}^2, d_{i1}d_{i2}, \dots, d_{i(m_3-1)}d_{im_3}, d_{i1}d_{i(m_3+1)}, \dots, d_{im_3}d_{i(m_3+m_2)})$.

Figure 16.3 shows some candidate designs generated in Step 2 for an MLFOD for four three-level factors and four two-level factors.

16.4 Results and Discussion

Table 16.1 provides $d_1, d_2,$ and r_{max} of 69 D-efficient ADSDs and two sets of corresponding new MLFODs with $m_3 = 4, \dots, 12, m_2 = 1, \dots, m_3,$ and $n \geq 16$. The goodness statistics are the first-order D-efficiency and the second-order D-efficiency, the maximum in terms of the absolute value of the correlation coefficients among $2m_3 + m_2$ columns of the model matrix \mathbf{X} for the pure quadratic model, respectively. The D-efficiencies d_1, d_2 of ADSDs are calculated according to Eqs. (16.5) and (16.8) in the Appendix. The first set of MLFODs labeled MLFOD₁ was obtained by selecting the first $m_3 + m_2$ columns of a conference matrix and then change the 0's to 1's in the last m_2 columns. The second set of MLFODs labeled MLFOD₂ was constructed by the ESA in Sect. 16.3 using $x = \lceil \frac{m}{5} \rceil$.

The advantage of the MLFODs over the orthogonal arrays (for the same number of three-level and two-level factors) is that the former require much less runs. At the same time, the former, unlike the latter, could guarantee that (i) all quadratic effects are orthogonal to main effects and (ii) all main effects are orthogonal to two-factor interactions. While orthogonality does not help in simplifying the data analysis which is now done entirely by computers, they help in the interpretations of the results which is the aim of the experimenters.

ADSs, due to their method of construction, always have two runs more than the corresponding new MLFODs. It can be seen in Table 16.1 that all ADSs have higher d_1 's than the corresponding MLFOD₂'s (but smaller d_1 's than the corresponding MLFOD₁'s). At the same time, nearly all MLFOD₂'s have higher d_2 's than the corresponding ADSs. The r_{max} 's of the ADSs are always higher than the ones of MLFODs. This is due to the fact that the correlation between any two quadratic effect columns is $\frac{1}{2} - \frac{2}{n-4}$ (see JN p. 129). This value approaches 1/2 as n becomes large. In table 16.1, d_1 (or d_2) values of MLFODs printed in bold are higher than the ones of ADSs for the same set of (m_3, m_2) . r_{max} values of MLFODs printed in bold are smaller than the ones of ADSs for the same set of (m_3, m_2) .

With the exception of the ADS for $m_3 = m_2 = 9$ and four ADSs constructed from a C matrix of order 22 (see Table 16.1), all ADSs in Table 16.1 have the f values reaching their lower bound. This fact guarantees that the constructed ADSs will have minimum correlations between a two-level factor and a three-level factor and between any two two-level factors.

Unlike ADSs, our MLFOD₂'s do not guarantee zero correlation among three-level factors. At the same time, unlike our MLFOD₂'s with $n = 16, 24, 32, 40,$ and 48 (whose D matrices were constructed from the Hadamard matrices or Plackett-Burman designs), ADSs do not have zero correlation among the two-level factors.

16.5 An Industrial Case Study

Lin and Kacker [7] describe an experiment aiming to improve the quality and productivity of wave soldering of circuit pack assemblies (CPA). This experiment, summarized in a study by Kenett et al. [6] (p. 474), has four yield variables and 17 factors (controllable variables). The four yield variables are: (i) Insulation resistance, (ii) Cleaning characterization, (iii) Soldering efficiency, and (iv) Solder mask cracking. The 17 factors are: (A) Type of activator, (B) Amount of activator, (C) Type of surfactant, (D) Amount of surfactant, (E) Amount of antioxidant, (F) Type of solvent, (G) Amount of solvent, (H) Amount of flux, (I) Preheat time, (J) Solder temperature, (K) Conveyor speed, (L) Conveyor angle, (M) Wave height setting, (N) Detergent concentration, (O) Detergent

Table 16.1 Comparison of D-efficiencies and r_{\max} of MLFODs and ADSDs

m_3	m_2	n^a	MLFOD ₁			MLFOD ₂			ADSD		
			d_1	d_2	r_{\max}	d_1	d_2	r_{\max}	d_1	d_2	r_{\max}
4	3	16	0.909	0.443	0.143	0.831	0.484	0.333	0.858	0.478	0.357
5	2	16	0.899	0.390	0.143	0.795	0.412	0.333	0.836	0.426	0.357
6	1	16	0.892	0.348	0.143	0.764	0.352	0.333	0.818	0.386	0.357
4	4	16	0.911	0.469	0.143	0.839	0.508	0.333	0.862	0.502	0.357
5	3	16	0.900	0.415	0.143	0.797	0.432	0.333	0.843	0.450	0.357
6	2	16	0.892	0.371	0.143	0.744	0.364	0.333	0.826	0.408	0.357
7	1	16	0.888	0.331	0.143				0.812	0.376	0.357
5	4	20	0.902	0.414	0.200	0.822	0.452	0.375	0.882	0.455	0.389
6	3	20	0.907	0.375	0.200	0.811	0.411	0.375	0.869	0.411	0.389
7	2	20	0.910	0.342	0.200	0.803	0.359	0.375	0.856	0.376	0.389
8	1	20	0.911	0.312	0.111	0.784	0.314	0.375	0.845	0.347	0.389
5	5	20	0.888	0.430	0.200	0.814	0.466	0.375	0.884	0.475	0.389
6	4	20	0.896	0.392	0.200	0.804	0.425	0.375	0.873	0.431	0.389
7	3	20	0.903	0.359	0.200	0.795	0.373	0.375	0.861	0.395	0.389
8	2	20	0.907	0.329	0.200	0.785	0.330	0.375	0.850	0.365	0.389
9	1	20	0.909	0.300	0.111	0.768	0.293	0.375	0.841	0.340	0.389
6	5	24	0.939	0.403	0.091	0.806	0.481	0.333	0.900	0.434	0.409
7	4	24	0.933	0.366	0.091	0.793	0.443	0.333	0.890	0.396	0.409
8	3	24	0.929	0.335	0.091	0.764	0.406	0.333	0.881	0.365	0.409
9	2	24	0.926	0.309	0.091	0.738	0.370	0.333	0.872	0.339	0.409
10	1	24	0.924	0.285	0.091	0.714	0.339	0.333	0.865	0.316	0.409
6	6	24	0.940	0.422	0.091	0.811	0.496	0.333	0.903	0.452	0.409
7	5	24	0.934	0.384	0.091	0.791	0.456	0.333	0.893	0.413	0.409
8	4	24	0.930	0.352	0.091	0.767	0.419	0.333	0.884	0.381	0.409
9	3	24	0.926	0.325	0.091	0.743	0.383	0.333	0.876	0.354	0.409
10	2	24	0.924	0.300	0.091	0.722	0.352	0.333	0.869	0.331	0.409
11	1	24	0.923	0.276	0.091	0.696	0.322	0.333	0.863	0.312	0.409
7	6	28	0.923	0.381	0.143	0.816	0.463	0.273	0.912	0.416	0.423
8	5	28	0.927	0.351	0.143	0.803	0.432	0.273	0.905	0.383	0.423
9	4	28	0.930	0.326	0.143	0.786	0.406	0.273	0.898	0.354	0.423
10	3	28	0.933	0.303	0.143	0.771	0.382	0.273	0.892	0.331	0.423
11	2	28	0.934	0.283	0.143	0.757	0.361	0.273	0.886	0.310	0.423
12	1	28	0.934	0.265	0.077	0.744	0.340	0.273	0.881	0.292	0.423
7	7	28	0.917	0.395	0.143	0.808	0.472	0.273	0.913	0.431	0.423
8	6	28	0.921	0.365	0.143	0.798	0.442	0.273	0.907	0.398	0.423
9	5	28	0.925	0.339	0.143	0.780	0.416	0.273	0.901	0.369	0.423
10	4	28	0.929	0.317	0.143	0.766	0.391	0.273	0.895	0.345	0.423
11	3	28	0.931	0.296	0.143	0.753	0.370	0.273	0.889	0.323	0.423
12	2	28	0.932	0.277	0.143	0.743	0.350	0.273	0.884	0.305	0.423
8	7	32	0.954	0.374	0.067	0.857	0.467	0.231	0.923	0.400	0.433
9	6	32	0.951	0.345	0.067	0.835	0.432	0.231	0.917	0.371	0.433
10	5	32	0.948	0.321	0.067	0.815	0.402	0.231	0.912	0.345	0.433
11	4	32	0.945	0.299	0.067	0.795	0.375	0.231	0.906	0.323	0.433
12	3	32	0.944	0.281	0.067	0.778	0.351	0.231	0.902	0.304	0.433
8	8	32	0.955	0.388	0.067	0.860	0.479	0.231	0.925	0.414	0.433
9	7	32	0.951	0.359	0.067	0.834	0.443	0.231	0.919	0.384	0.433
10	6	32	0.948	0.334	0.067	0.814	0.412	0.231	0.914	0.358	0.433
11	5	32	0.946	0.312	0.067	0.793	0.385	0.231	0.909	0.336	0.433
12	4	32	0.944	0.293	0.067	0.778	0.361	0.231	0.904	0.316	0.433
9	8	36	0.939	0.357	0.111	0.845	0.447	0.200	0.929	0.386	0.441

(continued)

Table 16.1 (continued)

m_3	m_2	n^a	MLFOD ₁			MLFOD ₂			ADSD		
			d_1	d_2	r_{\max}	d_1	d_2	r_{\max}	d_1	d_2	r_{\max}
10	7	36	0.941	0.333	0.111	0.840	0.422	0.200	0.925	0.359	0.441
11	6	36	0.943	0.312	0.111	0.832	0.397	0.200	0.921	0.336	0.441
12	5	36	0.945	0.293	0.111	0.827	0.373	0.200	0.917	0.316	0.441
9	9	36	0.935	0.368	0.111	0.844	0.456	0.200	0.929	0.398	0.441
10	8	36	0.938	0.344	0.111	0.838	0.431	0.200	0.926	0.371	0.441
11	7	36	0.940	0.323	0.111	0.829	0.406	0.200	0.922	0.348	0.441
12	6	36	0.942	0.304	0.111	0.821	0.381	0.200	0.919	0.328	0.441
10	9	40	0.963	0.351	0.053	0.867	0.440	0.216	0.937	0.374	0.447
11	8	40	0.961	0.328	0.053	0.857	0.414	0.216	0.933	0.350	0.447
12	7	40	0.959	0.308	0.053	0.844	0.390	0.216	0.930	0.329	0.447
10	10	40	0.963	0.363	0.053	0.867	0.450	0.216	0.938	0.385	0.447
11	9	40	0.961	0.339	0.053	0.858	0.424	0.216	0.935	0.361	0.447
12	8	40	0.959	0.319	0.053	0.845	0.400	0.216	0.931	0.339	0.447
11	10	44	0.946	0.337	0.095	0.841	0.460	0.222	0.935	0.361	0.452
12	9	44	0.946	0.317	0.095	0.832	0.438	0.222	0.931	0.339	0.452
11	11	44	0.944	0.347	0.095	0.842	0.469	0.222	0.933	0.370	0.452
12	10	44	0.943	0.326	0.140	0.830	0.446	0.222	0.932	0.349	0.452
12	11	48	0.969	0.333	0.043	0.874	0.457	0.200	0.947	0.353	0.457
12	12	48	0.969	0.343	0.043	0.877	0.466	0.200	0.948	0.362	0.457

^aRun size of MLFODs. For the same set of (m_3, m_2) ADSD requires two extra runs

temperature, (P) Cleaning conveyor speed (Q) Rinse water temperature. Out of these 17 factors, 7 factors (A), (C), (F), (M), (N), (O), and (Q) are two-level factors and the rest are three-level factors. The aim of this experiment is to single out the active factors and then apply a full quadratic model in these factors.

Let us consider two candidate MLFODs for ten three-level factors and eight two-level factors (including a blocking factor): (a) a 36-run MLFOD constructed by the ESA and (b) a 38-run ADSD. Both designs were constructed from a conference matrix of size 18. We do not consider the 30-run mixed-level screening design of [10] and the orthogonal arrays (http://support.sas.com/techsup/technote/ts723_Designs.txt) for the same number of three- and two-level factors as they are not MLFODs and as such might not possess the advantages of an MLFOD, namely, (i) all quadratic effects are orthogonal to main effects and (ii) all main effects are orthogonal to two-factor interactions. Besides, a 72-run orthogonal array exceeds the available budget for this experiment. Figure 16.4 displays the **D** matrices of two mentioned candidate MLFODs.

Table 16.1 shows the goodness statistics of the two candidate designs. The $d_1, d_2,$ and r_{\max} of the 38-run ADSD are 0.926, 0.371, and 0.441, and the one of the corresponding 36-run MLFOD are 0.838, 0.410, and 0.2. While the 38-run ADSD is superior to the 36-run MLFOD in terms of d_1 , it is inferior to the latter in terms of d_2 and r_{\max} . This pattern can be observed in Table 16.1 for all MLFOD₂'s with $n \geq 24$.

The correlation cell plots of the two candidate designs are shown in Fig. 16.5. These plots, used in Jones and

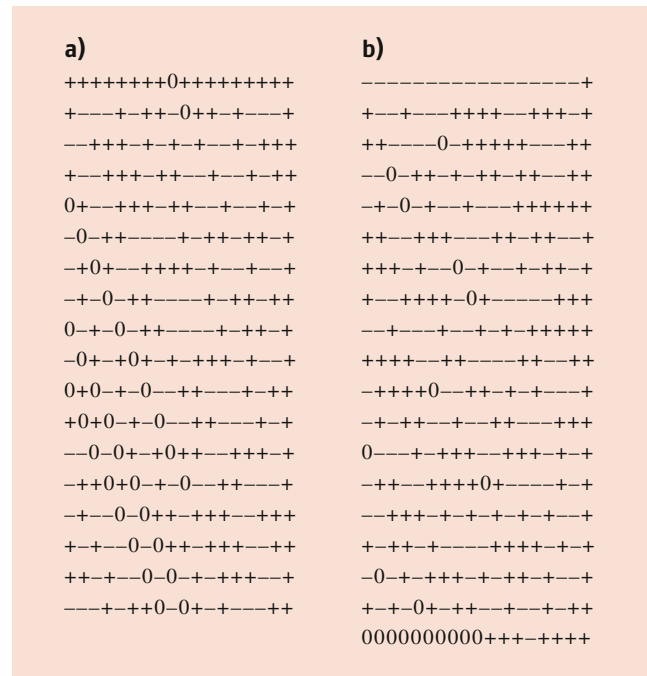


Fig. 16.4 The **D** matrices of two candidate MLFODs for ten three-level factors and eight two-level factors (including a blocking factor) for the PCA experiment described by Kenett et al. [6]: (a) of a 36-run MLFOD constructed by the ESA and (b) of a 38-run ADSD

Nachtsheim [3], display the magnitude of the correlation between main effects, quadratic effects of three-level factors and two-factor interactions in screening designs. The color of each cell in these plots goes from white (no correlation)

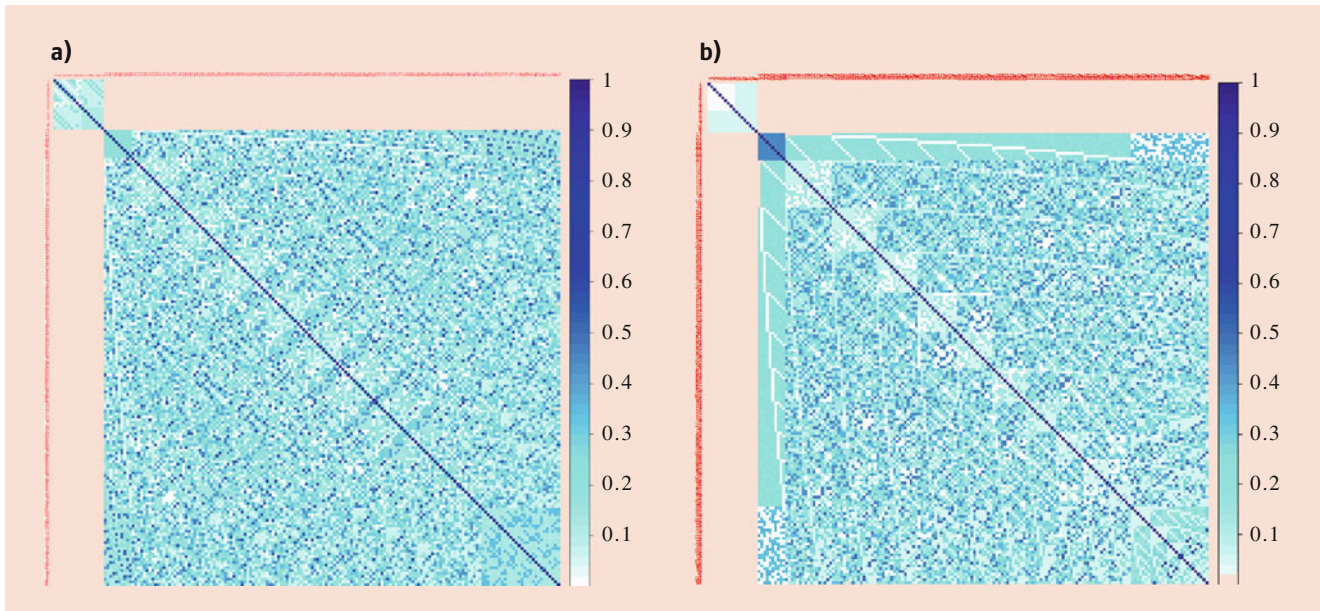


Fig. 16.5 Correlation cell plots of (a) 36-run MLFOD constructed by the ESA and (b) a 38-run ADSD for the experiment with ten three-level and eight two-level factors (including a blocking factor)

to dark (correlation of 1 or close to 1). Figure 16.5 confirms that for both designs, the main effects are orthogonal to the quadratic effects and two-factor interactions. It can be seen in Fig. 16.5b that while the main effects of three-level factors have zero correlation, the quadratic effects of these factors have fairly high correlation.

Both algorithms in Sects. 16.2 and 16.3 appear to be very fast and do not seem to be affected by the curse of dimensionality for design optimization. Both algorithms construct the abovementioned 38-run ADSD and 36-run MLFOD for ten three-level factors and eight two-level factors in less than 1 s on an HP EliteBook 8770w laptop. Note that the CEA uses 10,000 tries and out of 10,000 tries, 366 have the f values reaching the lower bound and all of them have the same values of (d_1, d_2, r_{\max}) .

16.6 Conclusion

Screening designs precede efforts to optimize a product or process. Their goal is to reduce a long list of factors so that the optimization effort can focus on a shorter list of factors. The literature on screening experiments in engineering and science popularized experiments with two-level factors such as orthogonal FFDs [1]. This constraint slows down the knowledge acquisition process by producing information only on main effects and interactions. In mixed-level screening designs, one combines factors at two and three levels, thus also producing information on quadratic effects. This paper presents two new classes of MLFODs. Both classes are economical and efficient. These designs provide more choices for the experimenters and help them to “design for

experiments instead of experiment for the design.” A wider scope to this work is the augmentation of data collected in nonexperimental contexts with experimentally designed add-on observations. The analysis of observed data provides initial information on the X space characteristics. To achieve the optimized conditions, the designs discussed in this paper can be used to expand the initial data sets. This approach was mentioned by Kenett and Nguyen [5] with examples of tools used to assess the X space statistical properties. In summary, we aim here at expanding the screening-optimization continuum with more flexible and optimized designs.

The zipped file containing the \mathbf{D} matrices (i.e., half fractions) for the designs in Table 16.1 and the input matrices we use to construct these \mathbf{D} matrices as well as the Java programs implementing the algorithms in Sects. 16.2 and 16.3 is downloadable from <https://drive.google.com/open?id=16thajCMzr4WLib-GkyKrBv2HHnziAIMr>.

Appendix: Calculating (d_1, d_2) Values of an MLFOD

Recall that $\mathbf{D}_{m^* \times (m_3 + m_2)} (= (d_{ij}))$ in (16.2) is the submatrix from which an MLFOD for m_3 three-level factors and m_2 two-level factors can be constructed from a matrix of order m ($m_3 + m_2 \leq m \leq m^*$). For ADSDs $m^* = m + 1$ and for our MLFODs $m^* = m$. The first-order D-efficiency d_1 and the pure quadratic D-efficiency d_2 of this MLFOD can be calculated as:

$$|\mathbf{X}'\mathbf{X}|^{1/p}/n \quad (16.3)$$

where \mathbf{X} , p , and n are the model matrix, the number of parameters for the models, and the number of runs, respectively.

For the first-order model, $p = 1 + m_3 + m_2$ and the i th row of \mathbf{X} can be written as $(1, d_{i1}, \dots, d_{i(m_3+m_2)})$. Thus the (information matrix) $\mathbf{X}'\mathbf{X}$ will be of the form

$$2 \begin{pmatrix} m^* & \mathbf{0}' \\ \mathbf{0} & \mathbf{B} \end{pmatrix}, \quad (16.4)$$

where $\mathbf{0}_{m_3+m_2}$ is a column vector of 0's and $\mathbf{B}_{(m_3+m_2) \times (m_3+m_2)} = \mathbf{D}'\mathbf{D}$. The determinant of $\mathbf{X}'\mathbf{X}$ for the first-order model can now be calculated as

$$|\mathbf{X}'\mathbf{X}| = 2^{1+m_3+m_2} m^* |\mathbf{B}|. \quad (16.5)$$

For the pure quadratic model, $p = 1 + m_3 + m_3 + m_2$ and the i th row of \mathbf{X} can be written as $(1, d_{i1}^2, \dots, d_{im_3}^2, d_{i1}, \dots, d_{i(m_3+m_2)})$. Thus, the matrix $\mathbf{X}'\mathbf{X}$ will be of the form

$$2 \begin{pmatrix} \mathbf{A} & \mathbf{0}' \\ \mathbf{0} & \mathbf{B} \end{pmatrix}, \quad (16.6)$$

where $\mathbf{0}_{(m_3+m_2) \times (1+m_3)}$ a matrix of 0's and $\mathbf{A}_{(1+m_3) \times (1+m_3)}$ is a matrix of the form

$$\begin{pmatrix} m^* & b\mathbf{1}' \\ b\mathbf{1} & \mathbf{A}^* \end{pmatrix} \quad (16.7)$$

assuming each of the m_3 three-level columns of \mathbf{D} has a fixed number b of ± 1 's. Here $\mathbf{1}_{m_3}$ is a column vector of 1's and $\mathbf{A}_{m_3 \times m_3}^*$ the core of \mathbf{A} in (16.7), i.e., the matrix \mathbf{A} without its first row and first column. The determinant of $\mathbf{X}'\mathbf{X}$ for the pure quadratic model can now be calculated as

$$|\mathbf{X}'\mathbf{X}| = 2^{1+2m_3+m_2} m^* |\mathbf{A}^* - \frac{b^2}{m^*} \mathbf{J}| |\mathbf{B}|. \quad (16.8)$$

For some MLFODs such as ADSDs, MLFOD₁, and MLFOD₂ with $n = 16$ and 38, the matrix \mathbf{A}^* in (16.7) will be of the form $c\mathbf{J} + d\mathbf{I}$ where \mathbf{I} is the identity matrix and \mathbf{J} is a matrix of 1's. Nguyen et al. [12] denoted this class of MLFODs as MLFOD*s. In these cases, $\mathbf{A}^* - \frac{b^2}{m^*} \mathbf{J}$ will also be of the form $c\mathbf{J} + d\mathbf{I}$. The determinants of a matrix of this form can be calculated as $d^{m_3} (1 + \frac{c}{d} \frac{m_3}{m})$. For ADSDs, $c = m - 2 - \frac{(m-1)^2}{m+1}$ and $d = 1$. For MLFOD₁, $c = m - 2 - \frac{(m-1)^2}{m}$ and $d = 1$.

References

1. Box, G.E., Hunter J.S., Hunter W.G.: Statistics for Experimenters: Design, Innovation, and Discovery, 2nd edn. John Wiley & Sons Ltd., New Jersey (2005)
2. Hedayat, A.S., Wallis, W.D.: Hadamard Matrices and Their Applications. *Ann. Stat.* **6**, 1184–1238 (1978)
3. Jones, B., Nachtsheim, C.J.: A class of three levels designs for definitive screening in the presence of second-order effects. *J. Qual. Technol.* **43**, 1–15 (2011)
4. Jones, B., Nachtsheim, C.J.: Definitive screening designs with added two-level categorical factors. *J. Qual. Technol.* **45**, 121–129 (2013)
5. Kenett R.S., Nguyen, N.K.: Experimental Learning: generate high information quality by comparing alternative experimental designs. *Qual. Prog.* **2017**, 40–47 (2017)
6. Kenett, R.S., Zacks, S., Amberti, D.: Modern Industrial Statistics: with applications in R, MINITAB and JMP, 3rd edn. John Wiley & Sons Ltd., West Sussex (2021)
7. Lin, K.M., Kacker R.N.: Optimization of the wave soldering process. In: Dehand, K. (eds.) *Quality Control and Robust Design, and The Taguchi Method*, pp. 143–157. Wadsworth BrooksCole, Pacific (1989)
8. Myer, R.K., Nachtsheim, C.J.: The coordinate-exchange algorithm for constructing exact optimal experimental designs. *Technometrics* **37**, 60–69 (1995)
9. Nguyen, N.-K.: An algorithmic approach to constructing supersaturated designs. *Technometrics* **38**, 69–73 (1996)
10. Nguyen, N.-K., Pham, D.-T.: Small mixed-level screening designs with orthogonal quadratic effects. *J. Qual. Technol.* **48**, 405–414 (2016)
11. Nguyen, N.-K., Stylianou, S.: Constructing definitive screening designs using cyclic generators. *J. Stat. Theory Pract.* **7**, 713–724 (2013)
12. Nguyen, N.-K., Pham, D.-T., Vuong, M.P.: Constructing D-efficient mixed-level foldover designs using Hadamard matrices. *Technometrics* **62**, 48–56 (2020)
13. Plackett, R.L., Burman, J.P.: The design of optimum multifactorial experiments. *Biometrika* **33**, 305–25 (1946)
14. Xiao, L.L., Lin, D.K.J., Bai, F.S.: Constructing definitive screening designs using conference matrices. *J. Qual. Technol.* **44**, 2–8 (2012)



Dr Nam-Ky Nguyen got his PhD. at the Indian Agricultural Research Institute, New Delhi, India in 1983. He has been a seasoned statistician with over 20 years of diversified experience in tertiary-level teaching, consulting and research at various universities and research organizations in Australia, USA, Vietnam, Thailand, Cambodia, Nigeria and Ivory Coast. Accredited by the Statistical Society of Australia (ASAT) in 1998. Author of around 50+ publications in applied statistics/computing and collaborative research. Author of the Gendex DOE toolkit. This toolkit is currently in use at several universities and research organizations worldwide. Research interest includes design of experiments, statistical computing and statistical modelling.



Professor Ron Kenett is Chairman of the KPA Group, Israel and Senior Research Fellow at the Neaman Institute, Technion, Haifa, Israel. He is an applied statistician combining expertise in academic, consulting and business domains. He serves on the editorial board of several international journals and was awarded the 2013 Greenfield Medal by the Royal Statistical Society and, in 2018, the Box Medal by the European Network for Business and Industrial Statistics. Ron holds a BSc in Mathematics (with first class honors) from Imperial College, London University and a PhD in Mathematics from the Weizmann Institute of Science, Rehovot, Israel.



Mai Phuong Vuong PhD is a lecturer at the School of Applied Mathematics and Informatics, Hanoi University of Science and Technology. She holds a Bachelor in applied Mathematics and Informatics from Vietnam National University, Hanoi, and a PhD in GeoInformatics from TU Bergakademie Freiberg, Germany. Her current research areas are design of experiments and data science.



Dr Tung Dinh Pham is lecturer at the Vietnam National University – Hanoi University of Science, Vietnam. He is an applied statistician combining expertise in academic, consulting domains. Tung holds a BSc in Mathematics and PhD in Mathematical Statistics from Hanoi University of Science.



Censored Data Prediction Based on Model Selection Approaches

Tzong-Ru Tsai, Jyun-You Chiang, Shuai Wang, and Yan Qin

Contents

17.1	Introduction	315
17.2	Methods for Approximate Predictors	317
17.2.1	Approximate Maximum Likelihood Estimation.....	317
17.2.2	Approximate Maximum Likelihood Predictors.....	319
17.3	Three Model Selection Approaches	322
17.4	Monte Carlo Simulations	323
17.5	Illustrative Examples	325
17.6	Conclusions	329
	References	330

and the proposed three model selection approaches perform well when more than one candidate distributions are competing for the best underlying model. Finally, the proposed approaches are applied to three data sets. This chapter is based on Chiang et al. (Math Probl Eng, 3465909, 2018).

Keywords

Discrimination · Maximum likelihood estimate · Censored data · Prediction · Location-scale distribution

Abstract

The contribution of this chapter is to solve two common problems in analysis. The first contribution is that we propose two methods to predict censored data. The second contribution is to automatically select the most suitable distribution function instead of subjective judgment. In this chapter, we propose three approaches of model selection. To demonstrate our approach, two members in the location-scale family, the normal distribution and smallest extreme value distribution, are used as candidates to illustrate the best model competition for the underlying distribution via using the proposed prediction methods. According to the result of Monte Carlo simulations, model misspecification has impact on the prediction precision

17.1 Introduction

At present, the advanced manufacturing technology is continuously improving the quality of components. If all test components are to be observed for failure time, it will lead to long experiment time. In order to save testing time and sample resource, censoring schemes are often considered to implement life tests. The type I censoring scheme and type II censoring scheme are two popular censoring schemes based on the criteria of test time censoring and failure number censoring.

In this chapter, the focus is using the type II censoring scheme for predicting the censored data for reliability evaluation when a discriminant problem is considered. In the type II censoring scheme, we consider an experiment where n identical components are placed in the test simultaneously. Assuming that the r th component fails, the experiment would be terminated. Thus, the last $(n - r)$ components are censored. In many engineering applications, censored data are not allowed for implementing statistical methods to obtain information. For example, if practitioners hope to integrate the life test and design of experimental methods, such that manufacturers can introduce a new product into the market quickly. The major difficulty is to implement the design of experimental methods for censored data. Predicting

T.-R. Tsai
Department of Statistics, Tamkang University, New Taipei City, Taiwan
e-mail: tzongru@gms.tku.edu.tw

J.-Y. Chiang (✉) · S. Wang
School of Statistics, Southwestern University of Finance and Economics, Chengdu, China

Y. Qin
School of Foreign Languages for Business, Southwestern University of Finance and Economics, Chengdu, China

unobserved order statistics from samples is a way to understand more about the censored data. Once the censored data are predicted, the predicted information with uncensored data can be merged as a pseudo-complete data set. Then the design of experiment methods can be employed for the pseudo-complete data set. The purpose of predicting life length of $s^{\text{th}}(r < s \leq n)$ item is equivalent to the life length of a $(n - s + 1)$ -out-of- n system that was made up of n identical components with independent life lengths. When $s = n$, it is better known as the parallel system. For this issue, various methods have been developed to predict the censored data. In the paper of Kaminsky and Nelson [1], they provided interval and point prediction of order statistics. Fertig et al. [2] proposed Monte Carlo estimates of percentiles of the distribution to construct prediction intervals for samples from a Weibull or extreme value distribution. Kaminsky and Rhodin [3] obtained the maximum likelihood predictor (MLP) to predict the future order statistics for the normal distribution. Wu et al. [4] obtained prediction intervals of future order statistics for the Pareto distribution based on the five new pivotal quantities. Kundu and Raqab [5] proposed the Bayesian prediction for the two-parameter Weibull distribution. Panahi and Sayyareh [6] provided parameter estimation and prediction of order statistics for the Burr type XII distribution. Some of these predictions are complex, or demand complex statistical models. Therefore, these existing methods are not easy to use in the real world.

Many authors also try to simplify the difficulty of the formula, and Raqab [7] provided four modified MLPs (MMLPs) to predict the future order statistics for the normal distribution (ND). Yang and Tong [8] used MMLP method to predict type II censored data from factorial experiments. Chiang [9] provided another three MMLP procedures to predict type II censored data for the Weibull distribution. In his procedures, it is difficult to find the only root solution to the parameter estimation. According to the previous literature, the MMLP method only has simple estimation results under ND, and the prediction results under other distributions are still very complex. So the parameter estimation of MMLP loses the advantage over other commonly used distributions.

The second important problem in life testing experiments is the model selection based on the existing sample. In practical applications, many statistical distributions are much alike, especially in censored data, and the underlying distribution of product quality characteristics is usually unknown. Several candidate models may fit the data well in practical applications. However, the predictions based on different candidate models may lead to a significant difference. Therefore, correct choice of the underlying distribution is of great

importance and has long been studied by many authors. Dumonceaux and Antle [10] used ratio of maximized likelihood (RML) to discriminate between the lognormal and Weibull distributions. Kundu and Manglick [11] proposed a model selection to discriminate between the lognormal and gamma distributions. Kundu and Raqab [12] provided a model selection method to discriminate between the generalized Rayleigh and lognormal distribution. Yu [13] provided a misspecification analysis method for the ND and smallest extreme value distribution (SEV). Ashour and Hashish [14] proposed a numerical comparison study for using RML-procedure, S-procedure, and F-procedure in failure model discrimination. Pakyari [15] proposed diagnostic tools based on the likelihood ratio test and the minimum Kolmogorov distance method to discriminate between the generalized exponential, geometric extreme exponential, and Weibull distributions. Elsherpieny et al. [16] provided a model selection to discriminate the gamma and log-logistic distributions based on progressive type II censored data.

In order to demonstrate our approaches, we take the location-scale family of distributions as an example, due to the well-developed theory and inferential procedures for the location-scale family of distributions, the model discrimination within the location-scale family of distributions is particularly important and it has received much attention. The main purpose of this chapter is to solve these issues and provide simple and satisfactory estimators of parameters and predictors of future order statistics when the underlying model is unknown but belongs to the location-scale family. Specifically, the prediction process of this study for censoring data prediction is presented in Fig. 17.1.

This chapter is based on Chiang et al. [17] and the rest of this chapter is organized as follows. Section 17.2 presents materials and methods. In this section, statistical methods to obtain approximate predictors for type II right-censored variables are studied and two prediction methods are proposed to predict the type II right-censored data based on the AMLEs. The ND and SEV are considered as the candidate distributions in this study to compete for the best model for obtaining the predictors of type II right-censored variables. In Sect. 17.3, we provide three algorithms to implement the three proposed model selection approaches to deal with the discrimination problem when obtaining the predictors of type II right-censored variables based on the proposed methods. An intensive simulation study is conducted in Sect. 17.4 to evaluate the performance of the proposed approaches. Then, three examples are used to demonstrate the applications of the proposed methodologies in Sect. 17.5. Some concluding remarks are provided in Sect. 17.6. The notations used in this chapter are defined in Table 17.1.

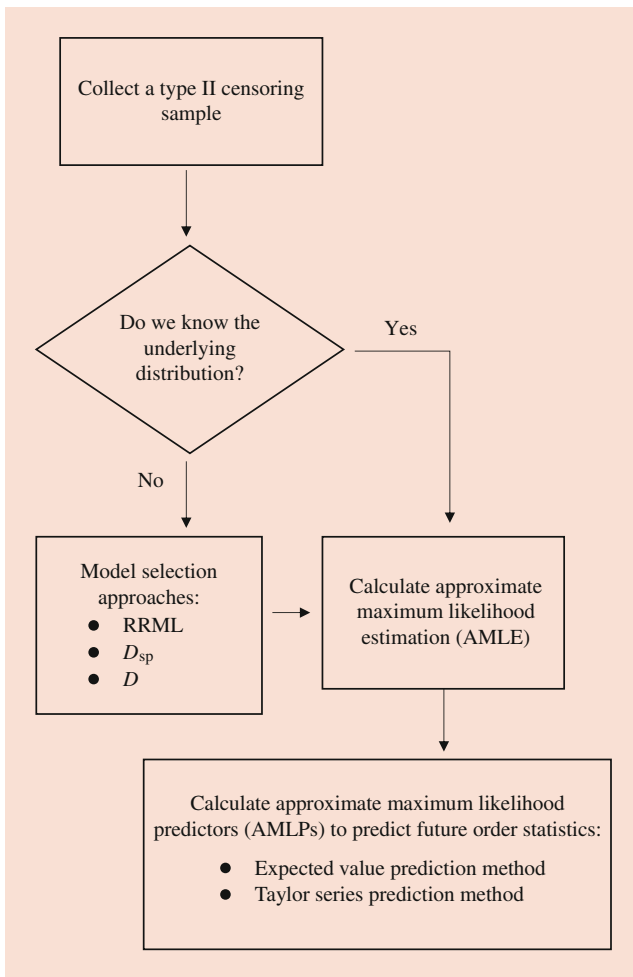


Fig. 17.1 The flow chart of the major contribution of this study

17.2 Methods for Approximate Predictors

17.2.1 Approximate Maximum Likelihood Estimation

Assume that random variable Y_i denotes the failure time of i th item and $X_i = \log(Y_i)$, which follows a location-scale family, having the probability density function (PDF) and cumulative distribution function (CDF):

$$f(x; \mu, \sigma) = \frac{1}{\sigma} g\left(\frac{x - \mu}{\sigma}\right), \quad (17.1)$$

and

$$f(x; \mu, \sigma) = G\left(\frac{x - \mu}{\sigma}\right), \quad -\infty < \mu < \infty, \sigma > 0, \quad (17.2)$$

$$-\infty < x < \infty,$$

respectively, where μ is location parameter and σ is scale parameter. $g(\cdot)$ and $G(\cdot)$ are the PDF and CDF of a member in the location-scale family. Let $\mathbf{x} = (x_{1:n}, x_{2:n}, \dots, x_{r:n})$ denote a type II censored sample and our goal is to predict $x_{s:n}$, where $1 \leq r < s \leq n$. Let $f(x) \equiv f(x; \mu, \sigma)$ and $F(x) \equiv F(x; \mu, \sigma)$ here and after to simplify the notations. Based on the paper of Kaminsky and Rhodin [3], they considered prediction of $X_{s:n}$ having observed. So the predictive likelihood functions (PLF) of $X_{s:n}$, μ and σ is

$$L(X_{s:n}, \mu, \sigma; \mathbf{x}) \equiv f(\mathbf{x}, X_{s:n}; \mu, \sigma) = \frac{n!}{(s-r-1)!(n-s)!} \prod_{j=1}^r f(x_{j:n}) [f(X_{s:n}) - f(x_{r:n})]^{s-r-1} f(X_{s:n}) [1 - f(X_{s:n})]^{n-s}. \quad (17.3)$$

Based on the proposed method by Raqab [7], the PLF of $X_{s:n}$, μ and σ in Eq. (17.3) can be represented as a product of two likelihood functions, the PLF of μ and σ (i.e., which is denoted as L_1) and the PLF of $X_{s:n}$ (i.e., which is denoted as L_2). Both likelihood functions are can be rewritten as

$$L_1(\mu, \sigma; \mathbf{x}) = \frac{n!}{(n-r)!} \prod_{j=1}^r f(x_{j:n}) [1 - f(x_{r:n})]^{n-r}, \quad (17.4)$$

and

$$L_2(X_{s:n}; \mu, \sigma, \mathbf{x}) = \frac{(n-r)!}{(s-r-1)!(n-s)!} \frac{[f(X_{s:n}) - f(x_{r:n})]^{s-r-1}}{[1 - f(x_{r:n})]^{n-r}} \times [1 - f(X_{s:n})]^{n-s} f(X_{s:n}). \quad (17.5)$$

In practice, we can obtain the MLEs of μ and σ , denoted by $\hat{\mu}$ and $\hat{\sigma}$, respectively, through maximizing $L_1(\mu, \sigma; \mathbf{x})$ in Eq. (17.4). Then using $\hat{\mu}$ and $\hat{\sigma}$ to replace μ and σ as the plug-in parameters in Eq. (17.5) to predict $X_{s:n}$. Let $z_{j:n} = (x_{j:n} - \mu)/\sigma$ for $j = 1, \dots, r$, $Z_{s:n} = (X_{s:n} - \mu)/\sigma$ for $s = r + 1, \dots, n$ and $\mathbf{z} = (z_{1:n}, z_{2:n}, \dots, z_{r:n})$, then the Eq. (17.4) and Eq. (17.5) can be rewritten as

$$L_1 \equiv L_1(\mu, \sigma; \mathbf{z}) = C_1 \prod_{j=1}^r \sigma^{-1} f(z_{j:n}) [1 - f(z_{r:n})]^{n-r} \quad (17.6)$$

and

$$L_2 \equiv L_2(Z_{s:n}; \hat{\mu}, \hat{\sigma}, \mathbf{z}) = C_2 \sigma^{-1} \frac{[f(Z_{s:n}) - f(z_{r:n})]^{s-r-1}}{[1 - f(z_{r:n})]^{n-r}} [1 - f(Z_{s:n})]^{n-s} f(Z_{s:n}), \quad (17.7)$$

Table 17.1 Notations and their descriptions

Notations	Descriptions
Y_i	The failure time of i th item of $Y, i = 1, \dots, n$
X_i	Take the logarithm of the Y_i
$f(x; \mu, \sigma), F(x; \mu, \sigma)$	The PDF and CDF of location-scale family with location parameter μ and scale parameter σ , respectively
$g(\cdot), G(\cdot)$	The standard PDF and CDF of location-scale family, respectively
$x_{i:n}$	The i th failure time of n samples of $x, i = 1, \dots, n$
r	The last sample number that can be observed
s	Sample number to prediction, $s = r + 1, \dots, n$
n	Sample size of x
\mathbf{x}	The type II censored sample
z	The standardized value of \mathbf{x}
$L(X_{s:n}, \mu, \sigma; \mathbf{x})$	The PLF of $X_{s:n}$
$L_1(\mu, \sigma; \mathbf{x})$ or L_1	The PLF of μ and σ
$L_2(X_{s:n}; \mu, \sigma, \mathbf{x})$ or L_2	The PLF of $X_{s:n}$
C_1, C_2	The constant in likelihood function
$\hat{\mu}, \hat{\sigma}$	The MLEs of μ and σ
$\tilde{\mu}, \tilde{\sigma}$	The AMLEs of μ and σ
$K, L, \lambda_1, \lambda_2, A_i, B_i, C_r, D_r$	The functions in the AMLE of μ and σ
$T'(\cdot), h'(\cdot), f'(\cdot), \phi'(\cdot)$	The first derivative of $T(\cdot), h(\cdot), f(\cdot)$ and $\phi(\cdot)$, respectively
$\Psi(\cdot), h(\cdot), h_1(\cdot)$	The functions in the log likelihood function
MLP _E , MLP _T	The predictors of $X_{s:n}$ based on the expected prediction method and the Taylor series prediction, respectively
$\hat{X}_{s:n}^{M,1}, \hat{X}_{s:n}^{M,2}$	The MLP _E and MLP _T of the $X_{s:n}$ under the candidate model M , respectively
ND, SEV	The abbreviations of the Normal and smallest extreme value distribution.
$\alpha, \beta, \gamma, \nu_s, \rho$	The functions in the MLP _T of the $X_{s:n}$ under the ND
$\alpha_s, \beta_s, \gamma_E, \nu_E, \rho_E$	The functions in the MLP _T of the $X_{s:n}$ under the SEV
M	The candidate model, $M = N$ (the ND) or SEV
$\hat{L}_{M_i}(\cdot), \hat{L}_N(\cdot), \hat{L}_S(\cdot)$	The PLF of $X_{s:n}$ based on the candidate model M , ND and SEV, respectively
$\hat{\mu}_N, \hat{\sigma}_N$	The MLEs of μ and σ based on the ND
$\phi(\cdot), \Phi(\cdot)$	The PDF and CDF of the standard ND, respectively
$\hat{\mu}_S, \hat{\sigma}_S$	The MLEs of μ and σ based on the SEV.
ϕ_{sev}, Φ_{sev}	The PDF and CDF of the standard SEV, respectively.
$\hat{L}_{A1}(\cdot)$	The maximum likelihood function value in all candidate models
$\hat{\mu}_{A1}, \hat{\sigma}_{A1}$	The MLEs of μ and σ in the distribution corresponding to $\hat{L}_{A1}(\cdot)$.
$\hat{X}_{s:n}^{A1,j}, \hat{X}_{s:n}^{A2,j}, \hat{X}_{s:n}^{A3,j}$	The best prediction value selected by three models respectively, $j = 1, 2$
RRML	The ratio of the maximized likelihood approach
$D_{sp}(\cdot)$	The modification of Michael statistic
$\hat{D}_{SP}^{A2}(\cdot)$	The smallest $D_{sp}(\cdot)$ in all candidate models
$\hat{\mu}_{A2}, \hat{\sigma}_{A2}$	The MLEs of μ and σ in the distribution corresponding to $\hat{D}_{SP}^{A2}(\cdot)$
$D(\cdot)$	The statistic of the Kolmogorov-Smirnov test
$\hat{D}^{A3}(\cdot)$	The smallest $D(\cdot)$ in all candidate models
$\hat{\mu}_{A3}, \hat{\sigma}_{A3}$	The MLEs of μ and σ in the distribution corresponding to $\hat{D}^{A3}(\cdot)$
$\hat{D}_{SP}^N(\cdot), \hat{D}_{SP}^{SEV}(\cdot)$	Evaluate the value of D_{sp} through using the ND and SEV
$\hat{D}^N(\cdot), \hat{D}^{SEV}(\cdot)$	Evaluate the value of D through using the ND and SEV
cp	The censoring proportion
correct(%)	The correct model selection rate
N	The Monte Carlo runs

where $C_1 = n!(n-r)!$ and $C_2 = (n-r)!/(s-r-1)!(n-s)!$. Then, we can obtain the log-likelihood equation as:

$$\frac{\partial \log(L_1)}{\partial \mu} = \frac{1}{\sigma} \left[\sum_{j=1}^r \Psi(z_{j:n}) + (n-r)h(z_{r:n}) \right] = 0 \quad (17.8)$$

$$\frac{\partial \log(L_1)}{\partial \sigma} = \frac{1}{\sigma} \left[-r + \sum_{j=1}^r \Psi(z_{j:n})z_{j:n} + (n-r)h(z_{r:n})z_{r:n} \right] = 0 \quad (17.9)$$

and

$$\frac{\partial \log(L_2)}{\partial Z_{s:n}} = (s-r-1)h_1(z_{r:n}, Z_{s:n}) - \Psi(Z_{s:n}) - (n-s)h(Z_{s:n}) = 0, \quad (17.10)$$

where

$$\Psi(Z_{j:n}) = -\frac{f'(Z_{j:n})}{f(Z_{j:n})}, j = 1, \dots, n, \text{ where } Z_{j:n} = z_{j:n} \text{ if } j \leq r, \quad (17.11)$$

$$h(Z_{j:n}) = \frac{f(Z_{j:n})}{1-f(Z_{j:n})}, j = 1, \dots, n, \text{ where } Z_{j:n} = z_{j:n} \text{ if } j \leq r, \quad (17.12)$$

and

$$h_1(z_{r:n}, Z_{s:n}) = \frac{f(Z_{s:n})}{f(Z_{s:n}) - f(z_{r:n})}. \quad (17.13)$$

We can see that the MLEs of μ and σ cannot be obtained in a closed form. Therefore, the MLEs of μ and σ need to be searched using numerical methods. To obtain proper initial solutions for implementing gradient computation methods, we consider using the AMLEs of μ and σ , denoted by $\tilde{\mu}$ and $\tilde{\sigma}$, respectively, from Hossain and Willan [18] as their initial solutions in this chapter. From the paper of Hossain and Willan [18], the AMLE of μ and σ as:

$$\tilde{\mu} = K - L\tilde{\sigma}, \quad (17.14)$$

and

$$\tilde{\sigma} = \frac{-\lambda_1 + \sqrt{\lambda_1^2 + 4r\lambda_2}}{2r}, \quad (17.15)$$

where

$$K = \frac{\sum_{i=1}^r B_i Z_{j:n} + (n-r)D_r Z_{r:n}}{\sum_{i=1}^r B_i + (n-r)D_r}$$

$$L = \frac{\sum_{i=1}^r A_i - (n-r)C_r}{\sum_{i=1}^r B_i + (n-r)D_r}$$

$$\lambda_1 = \sum_{i=1}^r (Z_{j:n} - K)A_i - (n-r)C_r(Z_{r:n} - K)$$

$$\lambda_2 = \sum_{i=1}^r (Z_{j:n} - K)^2 B_i + (n-r)D_r(Z_{r:n} - K)^2$$

$$A_i = T(v_{i:n}) - v_{i:n}T'(v_{i:n})$$

$$B_i = -T'(v_{i:n})$$

$$C_r = h(v_{r:n}) - v_{r:n}h'(v_{r:n})$$

and

$$D_r = h'(v_{r:n}),$$

where $T'(\cdot)$ and $h'(\cdot)$ are the first derivative of $T(\cdot)$ and $h(\cdot)$, respectively.

17.2.2 Approximate Maximum Likelihood Predictors

We propose two approximation prediction methods to predict $X_{s:n}$, the expected value prediction method and Taylor series prediction method. The resulting predictors of $X_{s:n}$ based on the expected prediction method are denoted by MLP_E , and the resulting predictors of $X_{s:n}$ based on the Taylor series prediction method are denoted by MLP_T . The two approximate methods mainly use two different methods to get the approximate $h_1(z_{r:n}, Z_{s:n})$ and $h(Z_{s:n})$.

The Expected Value Prediction Method

Based on the expected value prediction method, replacing (μ, σ) with $(\hat{\mu}, \hat{\sigma})$, and replacing $h_1(z_{r:n}, Z_{s:n})$ and $h(Z_{s:n})$ by their respective expected values in Eq. (17.10). According to Raqab [7], the expected values of $f(Z_{j:n})$, $h_1(z_{r:n}, Z_{s:n})$, and $h(Z_{s:n})$ can be presented, respectively, by

$$E[f(Z_{j:n})] = \frac{1}{n+1} \sum_{k=j+1}^{n+1} E[\Psi(Z_{k:n+1})], j \leq n \text{ and} \tag{17.16}$$

$$Z_{j:n} = z_{j:n} \text{ if } j \leq r,$$

$$E[h(Z_{j:n})] = \frac{1}{n-j} \sum_{k=j+1}^n E[\Psi(Z_{k:n})], j \leq n-1 \text{ and} \tag{17.17}$$

$$Z_{j:n} = z_{j:n} \text{ if } j \leq r,$$

and

$$E[h_1(Z_{i:n}, Z_{j:n})] = \frac{1}{j-i-1} \sum_{k=j}^n E[\Psi(Z_{k:n})], j-i \geq 2, \text{ and} \tag{17.18}$$

$$Z_{j:n} = z_{j:n} \text{ if } j \leq r.$$

We denote the MLP_E of the $X_{s:n}$ under the candidate model M by $\hat{X}_{s:n}^{M,1}$.

The Taylor Series Prediction Method

Based on the Taylor series prediction method, replacing (μ, σ) with $(\hat{\mu}, \hat{\sigma})$ and replacing $h(Z_{s:n})$ and $h_1(z_{r:n}, Z_{s:n})$ by their Taylor series approximations at the points $F^{-1}(p_s)$ and $(F^{-1}(p_r), F^{-1}(p_s))$, respectively, in Eq. (17.10). We denote the MLP_T of the $X_{s:n}$ under the candidate model M by $\hat{X}_{s:n}^{M,2}$.

In this study, we use ND and SEV as candidates to illustrating the applications of the proposed method. But the suggested algorithms in this study can be applied for the cases with more than two candidate members. The reason to select the ND and SEV as candidates is due to the Weibull distribution and lognormal distribution, which are two widely used distributions for life testing applications. The Weibull and lognormal distributions can be transformed into the SEV and ND, respectively, by taking log-transformation.

- *The normal distribution case:*

If the underlying distribution is normal, the PDF of normal distribution is given by

$$g(Z) = \phi(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2}. \tag{17.19}$$

According to Eq. (17.19), we can obtain $\Psi(z) = -\phi'(z)/\phi(z) = z$. The MLEs of ND parameters are denoted by $\hat{\mu}_N$ and $\hat{\sigma}_N$. Replacing μ and σ with $\hat{\mu}_N$ and $\hat{\sigma}_N$ in Eq. (17.6), the Eq. (17.6) can be rewritten as

$$\hat{L}_N(\hat{\mu}_N, \hat{\sigma}_N) = C_1 \prod_{j=1}^r \hat{\sigma}_N^{-1} \phi(z_{j:n}) [1 - \Phi(z_{r:n})]^{n-r}, \tag{17.20}$$

where $\Phi(\cdot)$ is the CDF of the standard ND. According to Eq. (17.17) and Eq. (17.18), $h_1(z_{r:n}, Z_{s:n})$ and $h(Z_{s:n})$ can be replaced by their respective expected values in Eq. (17.10). The Eq. (17.10) can be rewritten as

$$E(Z_{s:n}) - \hat{Z}_{s:n} = 0. \tag{17.21}$$

The value of $E(Z_{j:n})$ is available and has been tabulated by Teichroew [19]. Hence, the MLP_E of $X_{s:n}$ for ND can be derived as

$$\hat{X}_{s:n}^{N,1} = \hat{\mu}_N + \hat{\sigma}_N E(Z_{s:n}). \tag{17.22}$$

Because $E(Z_{s:n}) \geq z_{r:n}$ is a necessary condition, we modify Eq. (17.22) by

$$\hat{X}_{s:n}^{N,1} = \max\{\hat{\mu}_N + \hat{\sigma}_N E(Z_{s:n}), z_{r:n}\} \tag{17.23}$$

and use the $\hat{X}_{s:n}^{N,1}$ in Eq. (17.23) to protect $X_{s:n}$ for $r+1 \leq s \leq n$.

Based on the Taylor series prediction method, the functions $h(Z_{s:n})$ and $h_1(z_{r:n}, Z_{s:n})$ are expand by using the Taylor series around the points $F^{-1}(p_s)$ and $(F^{-1}(p_r), F^{-1}(p_s))$, respectively. According to Raqab [7], we can approximate $h(Z_{s:n})$ and $h_1(z_{r:n}, Z_{s:n})$ by

$$h(Z_{s:n}) = \frac{f(Z_{s:n})}{1-f(Z_{s:n})} \approx \alpha + \beta Z_{s:n}, \tag{17.24}$$

and

$$h_1(z_{r:n}, Z_{s:n}) = \frac{f(Z_{s:n})}{f(Z_{s:n}) - f(z_{r:n})} \approx \gamma + \rho z_{r:n} - \nu_s Z_{s:n}, \tag{17.25}$$

respectively, where

$$\alpha = \frac{f(\eta_s) \{(1 + \eta_s^2) q_s - \eta_s f(\eta_s)\}}{q_s^2},$$

$$\beta = \frac{f(\eta_s) \{f(\eta_s) - q_s \eta_s\}}{q_s^2},$$

$$\gamma = \frac{f(\eta_s) \{(1 + \eta_s^2) p_{sr} + \eta_s f(\eta_s) - \eta_r f(\eta_r)\}}{p_{sr}^2},$$

$$\rho = \frac{f(\eta_r) f(\eta_s)}{p_{sr}^2},$$

$$v_s = \frac{f(\eta_s) \{\eta_s p_{sr} + f(\eta_s)\}}{p_{sr}^2},$$

$$p_{ij} = p_i - p_j, p_i = i / (n + 1).$$

and

$\eta_i = F^{-1}(p_i)$ for $i = 1, 2, \dots, n$. The Eq. (17.10) can be rewritten by

$$(s - r - 1) (\gamma + \rho z_{r:n} - v_s Z_{s:n}) - z_{s:n} - (n - s) (\alpha + \beta Z_{s:n}) = 0. \tag{17.26}$$

The MLP_T of $X_{s:n}$ can be obtained by

$$\hat{X}_{s:n}^{N,2} = \max \left\{ \frac{(s - r - 1) \rho x_{r:n}}{(s - r - 1) v_s + 1 + (n - s) \beta} + \left[1 - \frac{(s - r - 1) \rho}{(s - r - 1) v_s + 1 + (n - s) \beta} \right] \hat{\mu}_N + \frac{(s - r - 1) \gamma - (n - s) \alpha}{(s - r - 1) v_s + 1 + (n - s) \beta} \hat{\sigma}_N, x_{r:n} \right\}, \tag{17.27}$$

where $r + 1 \leq s \leq n$.

• *The smallest extreme value distribution*

If the underlying distribution is SEV, the PDF of the SEV is given by

$$g(z) = \phi_{sev}(z) = e^{z - e^z}. \tag{17.28}$$

Based on the expected value prediction method, the $\Psi(z) = -\phi'_{sev}(z)/\phi_{sev}(z) = e^z - 1$. Using Eqs. (17.8) and (17.9), the MLEs of μ and σ are denoted by $\hat{\mu}_S$ and $\hat{\sigma}_S$, respectively. Replacing μ and σ with $\hat{\mu}_S$ and $\hat{\sigma}_S$ in Eq. (17.6), Eq. (17.6) can be represented by

$$\hat{L}_S(\hat{\mu}_S, \hat{\sigma}_S) = C_1 \prod_{j=1}^r \hat{\sigma}_S^{-1} \phi_{sev}(z_{j:n}) [1 - \Phi_{sev}(z_{r:n})]^{n-r}, \tag{17.29}$$

where $\Phi_{sev}(z) = 1 - \exp[-\exp(z)]$ is the CDF of the standard SEV. Then $h_1(z_{r:n}, Z_{s:n})$ and $h(Z_{s:n})$ are replaced by their respective expected values in Eq. (17.10). The Eq. (17.10) can be rewritten as

$$(s - r - 1) E[h_1(z_{r:n}, Z_{s:n})] - (e^{\hat{z}_{s:n}} - 1) - (n - s) E[h(Z_{s:n})] = 0. \tag{17.30}$$

The MLP_E of $X_{s:n}$ can be obtained as

$$\hat{X}_{s:n}^{SEV,1} = \max \{ \hat{\mu}_S + \hat{\sigma}_S \ln(E[\Psi(Z_{s:n})] + 1), x_{r:n} \} \tag{17.31}$$

for $r + 1 \leq s \leq n$ and $E\Psi(Z_{s:n}) = E(e^{Z_{s:n}} - 1)$.

Based on the Taylor series prediction method, expanding $h(Z_{s:n})$ and $h_1(z_{r:n}, Z_{s:n})$ by using the Taylor series at the points $F^{-1}(p_s)$ and $(F^{-1}(p_r), F^{-1}(p_s))$, respectively. We obtain

$$h(Z_{s:n}) = \frac{f(Z_{s:n})}{1 - f(Z_{s:n})} \approx 1 - \alpha_s - \beta_s Z_{s:n},$$

and

$$h_1(z_{r:n}, Z_{s:n}) = \frac{f(Z_{s:n})}{f(Z_{s:n}) - f(z_{r:n})} \approx \gamma_E + \rho_E z_{r:n} + v_E Z_{s:n}.$$

where

$$\alpha_s = 1 + \ln(q_s) - \ln(q_s) \ln(-\ln(q_s)),$$

$$\beta_s = \ln(q_s),$$

$$\gamma_E = q_s \ln(q_s) \{q_{rs} [-1 + (1 + \ln(q_s)) \ln(-\ln(q_s))] + q_s \ln(q_s) \ln(-\ln(q_s)) - q_r \ln(q_r) \ln(-\ln(q_r))\} / q_{rs}^2,$$

$$\rho_E = -q_s \ln(q_s) [(1 + \ln(q_s)) q_{rs} + q_s \ln(q_s)] / q_{rs}^2,$$

$$v_E = q_s \ln(q_s) q_r \ln(q_r) / q_{rs}^2,$$

and

$q_{ij} = q - q_j$. Eq. (17.10) can be rewritten as

$$(s - r - 1) (\gamma_E + \rho_E z_{r:n} + v_E Z_{s:n}) - e^{\hat{z}_{s:n}} - 1 - (n - s) (1 - \alpha_s - \beta_s Z_{s:n}) = 0 \tag{17.32}$$

The MLP_T of $X_{s:n}$ can be derived as

$$\hat{X}_{s:n}^{SEV,2} = \max \left\{ \frac{-(s-r-1)v_E X_{r:n}}{(s-r-1)\rho_E + \beta_s + (n-s)\beta_s} + \left[1 + \frac{(s-r-1)v_E}{(s-r-1)\rho_E + \beta_s + (n-s)\beta_s} \right] \hat{\mu}_S - \frac{(s-r-1)\gamma_E + \alpha_s - (n-s) + (n-s)\alpha_s}{(s-r-1)\rho_E + \beta_s + (n-s)\beta_s} \hat{\sigma}_S, X_{r:n} \right\}, \tag{17.33}$$

for $r + 1 \leq s \leq n$.

17.3 Three Model Selection Approaches

When we collect a data set, sometimes we cannot determine the best distribution of data. When the assumed distribution is incorrect, it may lead to incorrect results. Therefore, we suggest three approaches to discriminate the candidate distributions, the ratio of the maximized likelihood (RRML) approach, modification D_{SP} approach (shorted as D_{SP} approach), and modification D approach (shorted as the D approach), to obtain the predictor of $\hat{X}_{s:n}$. All these three approaches can be implemented to obtain the predictor of $X_{s:n}$ via using Algorithm 1 to Algorithm 3.

Algorithm 1: The RRML Approach

- Step 1: Collect a type II censoring sample, which has size n and r observed failure times, we consider k candidate distributions.
- Step 2: Obtain $(\hat{\mu}_{M_i}, \hat{\sigma}_{M_i})$ and $\hat{L}_{M_i}(\hat{\mu}_{M_i}, \hat{\sigma}_{M_i})$ for the candidate model $M_i, i = 1, 2, \dots, k$. Obtain the $X_{s:n}$ under the candidate model M_i and label it by $\hat{X}_{s:n}^{M_i, j}$ for $s = r + 1, \dots, n, i = 1, 2, \dots, k$ and $j = 1$ or 2 .
- Step 3: Let $\hat{X}_{s:n}^{A1, j}$ denote the predicted value of $X_{s:n}$ for $j = 1$ or 2 . Based on the method proposed by Dumonceaux and Antle [10], we can obtain $\hat{X}_{s:n}^{A1, j}$, which can provide the largest maximum likelihood information by

$$\hat{L}_{A1}(\hat{\mu}_{A1}, \hat{\sigma}_{A1}) = \max \left\{ \hat{L}_{M_1}(\hat{\mu}_{M_1}, \hat{\sigma}_{M_1}), \hat{L}_{M_2}(\hat{\mu}_{M_2}, \hat{\sigma}_{M_2}), \dots, \hat{L}_{M_k}(\hat{\mu}_{M_k}, \hat{\sigma}_{M_k}) \right\}$$

If the candidate distributions are ND and SEV, the Step 2 and Step 3 in the Algorithm 1 can be reduced to

- Step 2': Obtain $(\hat{\mu}_N, \hat{\sigma}_N), (\hat{\mu}_S, \hat{\sigma}_S), \hat{L}_N(\hat{\mu}_N, \hat{\sigma}_N)$ and $\hat{L}_S(\hat{\mu}_S, \hat{\sigma}_S)$. Obtain the $X_{s:n}$ under the ND ($\hat{X}_{s:n}^{N, j}$) and obtain the $X_{s:n}$ under the SEV ($\hat{X}_{s:n}^{SEV, j}$) for $s = r + 1, \dots, n$ and $j = 1$ or 2 .
- Step 3': Let $\hat{X}_{s:n}^{A1}$ denote the predicted value of $X_{s:n}$. Then

$$\hat{X}_{s:n}^{A1} = \begin{cases} \hat{X}_{s:n}^{N, j} & \text{if } \hat{L}_N(\hat{\mu}_N, \hat{\sigma}_N) > \hat{L}_S(\hat{\mu}_S, \hat{\sigma}_S) \\ \hat{X}_{s:n}^{SEV, j} & \text{, otherwise.} \end{cases} \text{ for } s = r + 1, \dots, n \text{ and } j = 1 \text{ or } 2.$$

Algorithm 2: The D_{SP} Approach

- Step 1: Collect a type II censoring sample, which has size n and r observed failure times.
- Step 2: Obtain $(\hat{\mu}_{M_i}, \hat{\sigma}_{M_i})$ for $i = 1, 2, \dots, k$, and then obtain $\hat{X}_{s:n}^{M_i, j}$ for $s = r + 1, \dots, n, i = 1, 2, \dots, k$ and $j = 1$ or 2 .
- Step 3: Based on the method proposed by Castro-Kuriss et al. [20], the modification of D_{SP} with censored observations can be presented by

$$D_{SP}(\mu, \sigma) = \max_{1 \leq i \leq r} \left\{ \frac{2}{\pi} \left| \arcsin \left(\sqrt{\frac{i-0.5}{n}} \right) - \arcsin \left(\sqrt{U_{i:n}} \right) \right| \right\}, \tag{17.34}$$

where $U_{i:n} = G\left(\frac{x_{i:n} - \mu}{\sigma}\right)$. The definition of $G(\cdot)$ is the same as that of Eq. (17.2); it represents the CDF of the assumed

distribution in model selection. Evaluate the value of D_{SP} through using the candidate model M_i for $i = 1, 2, \dots, k$.

Step 4: Let $\hat{X}_{s:n}^{A2, j}$ be the predicted value of $X_{s:n}$ for $j = 1$ or 2 , then $\hat{X}_{s:n}^{A2, j}$ can be obtained with the smallest \hat{D}_{SP} . That is, $\hat{X}_{s:n}^{A2, j}$ is the value corresponding to $\hat{D}_{SP}^{A2}(\hat{\mu}_{A2}, \hat{\sigma}_{A2})$, which is defined by

$$\hat{D}_{SP}^{A2}(\hat{\mu}_{A2}, \hat{\sigma}_{A2}) = \min \left\{ \hat{D}_{SP}(\hat{\mu}_{M_1}, \hat{\sigma}_{M_1}), \hat{D}_{SP}(\hat{\mu}_{M_2}, \hat{\sigma}_{M_2}), \dots, \hat{D}_{SP}(\hat{\mu}_{M_k}, \hat{\sigma}_{M_k}) \right\}.$$

If the candidate distributions are ND and SEV, the Step 2–Step 4 in the Algorithm 2 can be reduced to

- Step 2': Obtain $(\hat{\mu}_N, \hat{\sigma}_N)$ and $(\hat{\mu}_S, \hat{\sigma}_S)$. Obtain the $\hat{X}_{s:n}^{N, j}$ under the ND and obtain the $\hat{X}_{s:n}^{SEV, j}$ under the SEV distribution for $s = r + 1, \dots, n$ and $j = 1$ or 2 .
- Step 3': The modification of D_{SP} with censored observations can be presented by

$$D_{SP}(\mu, \sigma) = \max_{1 \leq i \leq r} \left\{ \frac{2}{\pi} \left| \arcsin \left(\sqrt{\frac{i-0.5}{n}} \right) - \arcsin \left(\sqrt{U_{i:n}} \right) \right| \right\},$$

where $U_{i:n} = G\left(\frac{x_{i:n}-\mu}{\sigma}\right)$. The definition of $G(\cdot)$ is the same as that of Eq. (17.2); it represents the CDF of the assumed

$$\hat{X}_{s:n}^{A2} = \begin{cases} \hat{X}_{s:n}^{Nj}, & \text{if } \hat{D}_{SP}^N(\hat{\mu}_N, \hat{\sigma}_N) < \hat{D}_{SP}^{SEV}(\hat{\mu}_S, \hat{\sigma}_S) \\ \hat{X}_{s:n}^{SEVj}, & \text{if } \hat{D}_{SP}^N(\hat{\mu}_N, \hat{\sigma}_N) \geq \hat{D}_{SP}^{SEV}(\hat{\mu}_S, \hat{\sigma}_S) \end{cases} \text{ for } s = r + 1, \dots, n \text{ and } j = 1 \text{ or } 2.$$

Algorithm 3: The D Approach

- Step 1: Collect a type II censoring sample, which has size n and r observed failure times.
- Step 2: Obtain $(\hat{\mu}_{M_i}, \hat{\sigma}_{M_i})$ for $i = 1, 2, \dots, k$, and then obtain $\hat{X}_{s:n}^{M_i j}$ for $s = r + 1, \dots, n, i = 1, 2, \dots, k$ and $j = 1$ or 2 .
- Step 3: Based on the method proposed by Castro-Kuriss et al. [20], the modification of $D(\mu, \sigma)$ with censoring observations can be presented by

$$D(\mu, \sigma) = \max_{1 \leq i \leq r} \left\{ \frac{2}{\pi} \left| \sqrt{\frac{i-0.5}{n}} - U_{i:n} \right| \right\} + \frac{0.5}{n}, \tag{17.35}$$

where $U_{i:n} = G\left(\frac{x_{i:n}-\mu}{\sigma}\right)$.

- Step 4: Let $\hat{X}_{s:n}^{A3j}$ be the predicted value of $X_{s:n}$ for $j = 1$ or 2 , then $\hat{X}_{s:n}^{A3j}$ can be obtained with the smallest $\hat{D}(\hat{\mu}_{M_i}, \hat{\sigma}_{M_i})$. That is, $\hat{X}_{s:n}^{A3j}$ is the value corresponding to $\hat{D}^{A3}(\hat{\mu}_{A3}, \hat{\sigma}_{A3})$, which is defined by

$$\hat{X}_{s:n}^{A3j} = \begin{cases} \hat{X}_{s:n}^{Nj}, & \text{if } \hat{D}^N(\hat{\mu}_N, \hat{\sigma}_N) < \hat{D}^{SEV}(\hat{\mu}_S, \hat{\sigma}_S) \\ \hat{X}_{s:n}^{SEVj}, & \text{if } \hat{D}^N(\hat{\mu}_N, \hat{\sigma}_N) \geq \hat{D}^{SEV}(\hat{\mu}_S, \hat{\sigma}_S) \end{cases} \text{ for } s = r + 1, \dots, n \text{ and } j = 1 \text{ or } 2.$$

17.4 Monte Carlo Simulations

In this section, the performance of the proposed three approaches with two predicting methods is investigated. The ND and SEV are considered as the candidate distributions for competing the best lifetime model in the simulation study. The data sets of type II censoring sample, $x_{1:n}, \dots, x_{r:n}$, used in the simulation were randomly generated from the ND

distribution in model selection. Evaluate the value of D_{SP} through using the normal and smallest extreme value distributions, and denoted them by $\hat{D}_{SP}^N(\hat{\mu}_N, \hat{\sigma}_N)$ and $\hat{D}_{SP}^{SEV}(\hat{\mu}_S, \hat{\sigma}_S)$, respectively.

- Step 4': Let $\hat{X}_{s:n}^{A2j}$ denote the predicted value of $X_{s:n}$, then $\hat{X}_{s:n}^{A2j}$ can be obtained by

$$\hat{D}^{A3}(\hat{\mu}_{A3}, \hat{\sigma}_{A3}) = \min \left\{ \hat{D}(\hat{\mu}_{M_1}, \hat{\sigma}_{M_1}), \hat{D}(\hat{\mu}_{M_2}, \hat{\sigma}_{M_2}), \dots, \hat{D}(\hat{\mu}_{M_k}, \hat{\sigma}_{M_k}) \right\}.$$

If the candidate distributions are ND and SEV, the Step 2, Step 3, and Step 4 in the Algorithm 3 can be reduced to

- Step 2': Obtain $(\hat{\mu}_N, \hat{\sigma}_N)$ and $(\hat{\mu}_S, \hat{\sigma}_S)$. Obtain the $\hat{X}_{s:n}^{Nj}$ under the ND and obtain the $\hat{X}_{s:n}^{SEVj}$ under the SEV for $s = r + 1, \dots, n$ and $j = 1$ or 2 .

- Step 3': The modification of $D(\mu, \sigma)$ with censoring observations can be presented by

$$D(\mu, \sigma) = \max_{1 \leq i \leq r} \left\{ \frac{2}{\pi} \left| \sqrt{\frac{i-0.5}{n}} - U_{i:n} \right| \right\} + \frac{0.5}{n},$$

where $U_{i:n} = G\left(\frac{x_{i:n}-\mu}{\sigma}\right)$. Evaluate the value of $D(\mu, \sigma)$ by using the ND and SEV, and denoted them by $\hat{D}^N(\hat{\mu}_N, \hat{\sigma}_N)$ and $\hat{D}^{SEV}(\hat{\mu}_S, \hat{\sigma}_S)$.

- Step 4': Let $\hat{X}_{s:n}^{A3j}$ denote the predicted value of $X_{s:n}$, then $\hat{X}_{s:n}^{A3j}$ can be obtained by

and SEV with location parameter $\mu = 0$ and scale parameter $\sigma = 1$. Then, the s th order statistic is predicted and denoted by $\hat{X}_{s:n}$ for $s = r + 1, r + 2, \dots, n$ for the sample sizes $n = 20, 40$, and 60 . In this simulation, we considered different censoring proportions ($cp = r/n$) $cp = 0.9, 0.8, 0.7, 0.6, 0.5$ and different predicting 100pth percentile ($cp < p \leq 1$). For the purpose of comparison, the values of the bias and mean square error (MSE) of $\hat{X}_{s:n}$ are evaluated with $N = 10000$ Monte Carlo runs:

Table 17.2 The corresponding bias and MSEs for different settings with model misspecification when true distribution is ND

			Assumed distribution							
			Normal distribution				Extreme value distribution			
			$\hat{X}_{s:n}^{N,1}$		$\hat{X}_{s:n}^{N,2}$		$\hat{X}_{s:n}^{SEV,1}$		$\hat{X}_{s:n}^{SEV,2}$	
<i>n</i>	cp	<i>p</i>	bias	MSE	bias	MSE	bias	MSE	bias	MSE
20	0.8	0.9	-0.0559	0.0912	-0.2224	0.1268	-0.1951	0.1157	-0.2709	0.1504
	0.7	0.8	-0.0494	0.0594	-0.1832	0.0860	-0.1335	0.0702	-0.2076	0.0955
	0.7	0.9	-0.0685	0.1309	-0.1485	0.1422	-0.2873	0.1989	-0.3216	0.2193
	0.6	0.7	-0.0459	0.0481	-0.1623	0.0710	-0.1032	0.0547	-0.1773	0.0763
	0.6	0.8	-0.0739	0.1036	-0.1345	0.1119	-0.2257	0.1417	-0.2579	0.1567
	0.6	0.9	-0.0999	0.1967	-0.1574	0.2052	-0.3965	0.3212	-0.4213	0.3406
	0.5	0.6	-0.0542	0.0456	-0.1585	0.0659	-0.0953	0.0501	-0.1694	0.0694
	0.5	0.7	-0.0644	0.0878	-0.1145	0.0946	-0.1832	0.1144	-0.2152	0.1271
	0.5	0.8	-0.0921	0.1497	-0.1322	0.1554	-0.3146	0.2292	-0.3358	0.2423
	0.5	0.9	-0.1277	0.2668	-0.1747	0.2741	-0.5038	0.4667	-0.524	0.4863
40	0.8	0.9	0.0776	0.0724	-0.0546	0.0494	-0.2573	0.1319	-0.2022	0.0932
	0.7	0.8	0.0696	0.0506	-0.0404	0.0305	-0.1829	0.0765	-0.1336	0.0498
	0.7	0.9	0.0744	0.0711	-0.0146	0.0545	-0.3102	0.171	-0.2323	0.1123
	0.6	0.7	0.0633	0.0419	-0.034	0.0243	-0.1543	0.0584	-0.1045	0.0364
	0.6	0.8	0.0691	0.0576	-0.0064	0.0407	-0.234	0.116	-0.1464	0.0654
	0.6	0.9	0.0799	0.0807	0.009	0.0671	-0.3082	0.1776	-0.2455	0.1309
	0.5	0.6	0.0524	0.0398	-0.0337	0.0232	-0.1507	0.0544	-0.0927	0.0324
	0.5	0.7	0.0628	0.0465	0.0032	0.0321	-0.1996	0.0947	-0.0963	0.0454
	0.5	0.8	0.0728	0.0591	0.0195	0.0482	-0.2273	0.1168	-0.1436	0.0717
	0.5	0.9	0.0768	0.0809	0.0235	0.0714	-0.2995	0.1728	-0.2438	0.1334
60	0.8	0.9	0.0762	0.0535	-0.0227	0.0356	-0.2778	0.1286	-0.1802	0.072
	0.7	0.8	0.0686	0.0389	-0.0143	0.0224	-0.1969	0.0723	-0.1095	0.0365
	0.7	0.9	0.0766	0.0507	0.0076	0.0391	-0.3142	0.1573	-0.2203	0.0936
	0.6	0.7	0.0631	0.0317	-0.0112	0.0176	-0.1673	0.0561	-0.0799	0.0256
	0.6	0.8	0.0697	0.0425	0.0111	0.0309	-0.2381	0.1069	-0.1341	0.0529
	0.6	0.9	0.0798	0.0598	0.0268	0.0506	-0.3148	0.1652	-0.2406	0.1153
	0.5	0.6	0.0538	0.0295	-0.0104	0.0172	-0.1664	0.0546	-0.0676	0.0234
	0.5	0.7	0.0692	0.0322	0.0212	0.0222	-0.2064	0.0846	-0.0842	0.0336
	0.5	0.8	0.0803	0.0413	0.0388	0.0337	-0.2302	0.1063	-0.1358	0.0592
	0.5	0.9	0.0894	0.0534	0.0453	0.0469	-0.3072	0.1608	-0.2448	0.1179

$$\text{bias} = \frac{1}{N} \sum_{i=1}^N (\hat{X}_{s:n,i} - X_{s:n})$$

and

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (\hat{X}_{s:n,i} - X_{s:n})^2,$$

where $\hat{X}_{s:n,i}$ is the predicted value of $X_{s:n}$ that is obtained in the i th iteration of simulation for $i = 1, \dots, N$. All simulation results are displayed in Tables 17.2 and 17.3 with the candidate distributions of ND and SEV. From Tables 17.2 and 17.3, we notice that:

1. The bias and MSE are large when the misspecification model is used.

2. The impact of misspecification depends on the values of r and s .
3. As n or r increases, the simulated bias and MSE decrease.
4. We also find that the MSE based on using the Taylor series prediction method is smaller than that based on using the expected values prediction method when the sample size is or larger than 40.

To evaluate the performance of the three proposed model selection approaches for MLP. Tables 17.4, 17.5, and 17.6 report the simulation results for three model selection approaches from the ND. Tables 17.7, 17.8, and 17.9, respectively, report the simulation results for three model selection approaches from the SEV. The column “correct(%)” presented in Tables 17.4, 17.5, 17.6, 17.7, 17.8, and 17.9 is the correct model selection rate in all simulation runs. From Tables 17.4, 17.5, 17.6, 17.7, 17.8, and 17.9, we find that:

Table 17.3 The corresponding bias and MSEs for different settings with model misspecification when the true distribution is SEV

			Assumed distribution							
			Normal distribution				Extreme value distribution			
			$\hat{X}_{s:n}^{N,1}$		$\hat{X}_{s:n}^{N,2}$		$\hat{X}_{s:n}^{SEV,1}$		$\hat{X}_{s:n}^{SEV,2}$	
<i>n</i>	<i>cp</i>	<i>p</i>	bias	MSE	bias	MSE	bias	MSE	bias	MSE
20	0.8	0.9	0.3244	0.2161	-0.0603	0.0588	-0.0318	0.0633	-0.1622	0.0837
	0.7	0.8	0.2185	0.1425	-0.0896	0.0546	-0.0323	0.0511	-0.1526	0.0719
	0.7	0.9	0.3428	0.2439	0.1161	0.0999	-0.0580	0.1011	-0.1205	0.1081
	0.6	0.7	0.1554	0.1142	-0.1068	0.0551	-0.0315	0.0474	-0.1516	0.0675
	0.6	0.8	0.2358	0.1802	0.0564	0.0891	-0.0621	0.0968	-0.1206	0.1025
	0.6	0.9	0.3805	0.3264	0.2218	0.2011	-0.0829	0.1660	-0.1282	0.1704
	0.5	0.6	0.1089	0.1028	-0.1238	0.0641	-0.0374	0.0534	-0.1596	0.0747
	0.5	0.7	0.1864	0.1518	0.0396	0.0913	-0.0549	0.1009	-0.1120	0.1075
	0.5	0.8	0.2897	0.2633	0.1697	0.1855	-0.0802	0.1709	-0.1187	0.1748
	0.5	0.9	0.4603	0.5008	0.3419	0.3814	-0.1115	0.2556	-0.1471	0.2586
40	0.8	0.9	0.5659	0.3576	0.1264	0.0432	0.0838	0.0487	-0.0235	0.0311
	0.7	0.8	0.4603	0.2426	0.0696	0.0263	0.0807	0.0432	-0.0214	0.0246
	0.7	0.9	0.5673	0.3535	0.2467	0.0897	0.0864	0.0472	0.0160	0.0339
	0.6	0.7	0.4053	0.2016	0.0423	0.0214	0.0741	0.0414	-0.0231	0.0218
	0.6	0.8	0.4595	0.2434	0.1558	0.0536	0.0790	0.0474	0.0118	0.0334
	0.6	0.9	0.5704	0.3647	0.3146	0.1367	0.0920	0.0564	0.0399	0.0459
	0.5	0.6	0.3854	0.1911	0.0264	0.0229	0.0697	0.0442	-0.0262	0.0243
	0.5	0.7	0.4169	0.1981	0.1172	0.0387	0.0828	0.0424	0.0181	0.0289
	0.5	0.8	0.4685	0.2452	0.2188	0.0754	0.0908	0.0472	0.0440	0.0369
	0.5	0.9	0.5746	0.3562	0.3594	0.1576	0.0982	0.0506	0.0582	0.0426
60	0.8	0.9	0.5740	0.3528	0.1625	0.0464	0.0754	0.0357	-0.0028	0.0238
	0.7	0.8	0.4607	0.2327	0.0933	0.0250	0.0694	0.0320	-0.0046	0.0187
	0.7	0.9	0.5751	0.3519	0.2741	0.0948	0.0788	0.0344	0.0261	0.0255
	0.6	0.7	0.4073	0.1872	0.0636	0.0189	0.0659	0.0302	-0.0040	0.0167
	0.6	0.8	0.4602	0.2324	0.1754	0.0518	0.0710	0.0346	0.0227	0.0259
	0.6	0.9	0.5740	0.3554	0.3363	0.1385	0.0809	0.0422	0.0426	0.0356
	0.5	0.6	0.3812	0.1740	0.0434	0.0189	0.0570	0.0318	-0.0080	0.0182
	0.5	0.7	0.4144	0.1865	0.1343	0.0338	0.0743	0.0311	0.0299	0.0215
	0.5	0.8	0.4702	0.2319	0.2349	0.0714	0.0836	0.0346	0.0506	0.0284
	0.5	0.9	0.5830	0.3496	0.3826	0.1611	0.0921	0.0363	0.0625	0.0313

1. The three model selection approaches have good ability to identify the correct underlying distribution with a high probability.
2. The MSEs of these three approaches are close to those simulated MSEs of the cases by using the real underlying distribution.
3. The correct model selection rates through using the D_{SP} approach or D approach are higher than that of using the RRML approach when the sample size is smaller than 40. When the sample size grows to or over 40, the performance of the RRML approach is improved and the correct model selection rate of the RRML approach is higher than that is obtained by using the D_{SP} or D approach.
4. To compare the performance of using two different MLPs, the MSEs of using the expected values prediction method

are smaller than that using the Taylor series prediction method when the sample size is smaller than 40.

5. The proposed approaches can perform well under large sample size cases.

17.5 Illustrative Examples

In this section, three numerical examples are presented to illustrate the proposed approaches in Sects. 17.2, 17.3, and 17.4.

Example 1

In the paper of Mann and Fertig [21], a test airplane component's failure time dataset is provided, in which 13 components were placed on test, and the test was terminated at the

Table 17.4 The corresponding bias and MSEs for different settings of RML approach when the true distribution is ND

RML approach							
n	cp	p	$\hat{X}_{s:n}^{A1,1}$		$\hat{X}_{s:n}^{A1,2}$		correct(%)
			bias	MSE	bias	MSE	
20	0.8	0.9	-0.1178	0.0965	-0.2419	0.1365	0.7204
	0.7	0.8	-0.0955	0.0636	-0.1961	0.0911	0.6963
	0.7	0.9	-0.1619	0.1584	-0.2159	0.1752	0.6963
	0.6	0.7	-0.0814	0.0513	-0.1720	0.0744	0.6665
	0.6	0.8	-0.1473	0.1212	-0.1884	0.1329	0.6657
	0.6	0.9	-0.2262	0.2529	-0.2644	0.2671	0.6657
	0.5	0.6	-0.0826	0.0484	-0.1665	0.0686	0.6477
	0.5	0.7	-0.1249	0.1009	-0.1609	0.1106	0.6477
	0.5	0.8	-0.1934	0.1886	-0.2202	0.1981	0.6477
	0.5	0.9	-0.2884	0.3606	-0.3199	0.3739	0.6477
40	0.8	0.9	0.0656	0.0693	-0.0585	0.0503	0.9726
	0.7	0.8	0.0608	0.0488	-0.0428	0.0309	0.9762
	0.7	0.9	0.0636	0.0701	-0.0202	0.0557	0.9762
	0.6	0.7	0.0575	0.0404	-0.0354	0.0245	0.9785
	0.6	0.8	0.0621	0.0564	-0.0094	0.0411	0.9777
	0.6	0.9	0.0713	0.0801	0.0037	0.0682	0.9777
	0.5	0.6	0.0474	0.0388	-0.0347	0.0234	0.9817
	0.5	0.7	0.0571	0.0459	0.0012	0.0323	0.9817
	0.5	0.8	0.0664	0.0588	0.0161	0.0487	0.9817
	0.5	0.9	0.0687	0.0808	0.0178	0.0723	0.9817
60	0.8	0.9	0.0709	0.0525	-0.0245	0.0360	0.9922
	0.7	0.8	0.0654	0.0380	-0.0152	0.0224	0.9927
	0.7	0.9	0.0733	0.0503	0.0058	0.0395	0.9927
	0.6	0.7	0.0607	0.0313	-0.0115	0.0177	0.9941
	0.6	0.8	0.0667	0.0422	0.0086	0.0310	0.9933
	0.6	0.9	0.0764	0.0597	0.0246	0.0510	0.9933
	0.5	0.6	0.0523	0.0292	-0.0107	0.0172	0.9928
	0.5	0.7	0.0673	0.0317	0.0203	0.0223	0.9928
	0.5	0.8	0.0785	0.0413	0.0378	0.0338	0.9928
	0.5	0.9	0.0869	0.0532	0.0435	0.0469	0.9928

Table 17.5 The corresponding bias and MSEs for different settings of D_{sp} approach when the true distribution is ND

D_{sp} approach							
n	cp	p	$\hat{X}_{s:n}^{A2,1}$		$\hat{X}_{s:n}^{A2,2}$		correct(%)
			bias	MSE	bias	MSE	
20	0.8	0.9	-0.1036	0.0933	-0.2374	0.1342	0.7323
	0.7	0.8	-0.0760	0.0609	-0.1906	0.0890	0.7206
	0.7	0.9	-0.1334	0.1500	-0.1987	0.1660	0.7206
	0.6	0.7	-0.0618	0.0490	-0.1665	0.0725	0.6943
	0.6	0.8	-0.1184	0.1149	-0.1711	0.1259	0.6935
	0.6	0.9	-0.1869	0.2357	-0.2355	0.2479	0.6935
	0.5	0.6	-0.0619	0.0459	-0.1602	0.0664	0.6781
	0.5	0.7	-0.0953	0.0946	-0.1428	0.1036	0.6781
	0.5	0.8	-0.1561	0.1719	-0.1926	0.1801	0.6781
	0.5	0.9	-0.2360	0.3249	-0.2765	0.3358	0.6781
40	0.8	0.9	-0.0293	0.0601	-0.0890	0.0583	0.7009
	0.7	0.8	-0.0155	0.0369	-0.0610	0.0339	0.6966
	0.7	0.9	-0.0345	0.0693	-0.0678	0.0679	0.6966
	0.6	0.7	-0.0095	0.0300	-0.0479	0.0263	0.7018
	0.6	0.8	-0.0154	0.0489	-0.0381	0.0453	0.7010
	0.6	0.9	-0.0270	0.0820	-0.0530	0.0812	0.7010
	0.5	0.6	-0.0085	0.0283	-0.0432	0.0243	0.7535
	0.5	0.7	0.0008	0.0381	-0.0142	0.0337	0.7535
	0.5	0.8	0.0012	0.0548	-0.0132	0.0525	0.7535
	0.5	0.9	-0.0126	0.0833	-0.0336	0.0824	0.7535
60	0.8	0.9	-0.0131	0.0473	-0.0541	0.0421	0.7529
	0.7	0.8	-0.0047	0.0299	-0.0331	0.0248	0.7452
	0.7	0.9	-0.0154	0.0535	-0.0402	0.0503	0.7452
	0.6	0.7	-0.0007	0.0234	-0.0234	0.0188	0.7643
	0.6	0.8	-0.0017	0.0383	-0.0182	0.0344	0.7635
	0.6	0.9	-0.0072	0.0647	-0.0263	0.0627	0.7635
	0.5	0.6	0.0024	0.0226	-0.0182	0.0178	0.7864
	0.5	0.7	0.0172	0.0277	0.0056	0.0237	0.7864
	0.5	0.8	0.0202	0.0401	0.0099	0.0371	0.7864
	0.5	0.9	0.0083	0.0605	-0.0087	0.0587	0.7864

time of the 10th failure. The failure times (in hours) of the 10 components that failed were

$$D_1 : 0.22, 0.50, 0.88, 1.00, 1.32, 1.33, 1.54, 1.76, 2.50, 3.00.$$

Suppose Y_1 is the logs of the ten observations, that is, $Y_1 = \ln(D_1)$. Figure 17.2 presents the histogram and the estimated PDFs of the ND and SEV distribution. From Fig. 17.2, it is difficult to full decide the best distribution for lifetime fitting, because both candidate distributions can provide good fitting for this data set. In this example, the D_{SP} approach is considered to discriminate competing models and use expected values prediction method and Taylor series prediction method to predict the future order statistics, which are censored.

Based on this censored data, the AMLEs of the model parameters via using the normal distribution and extreme value distribution are $(\tilde{\mu}_N, \tilde{\sigma}_N) = (0.474, 0.936)$ and $(\tilde{\mu}_{EV}, \tilde{\sigma}_{EV}) = (0.811, 0.708)$, respectively. Through using Newton-Raphson algorithm with the initial values of $(\tilde{\mu}_N, \tilde{\sigma}_N)$ and $(\tilde{\mu}_{EV}, \tilde{\sigma}_{EV})$, we obtained the MLEs of μ and σ as $(\hat{\mu}_N, \hat{\sigma}_N) = (0.479, 0.938)$ and $(\hat{\mu}_S, \hat{\sigma}_S) = (0.821, 0.705)$ for the ND and SEV, respectively.

The D_{SP} values via using ND and SEV are 0.223 and 0.212, respectively. We claim the best distribution of this data set is SEV. The Taylor series prediction and expected values prediction for $(Y_{11:13}, Y_{12:13}, Y_{13:13})$ under the extreme value distribution with the censored sample can be obtained by $(\hat{Y}_{11:13}^{A2,2}, \hat{Y}_{12:13}^{A2,2}, \hat{Y}_{13:13}^{A2,2}) = (1.098, 1.281, 1.567)$ and $(\hat{Y}_{11:13}^{A2,1}, \hat{Y}_{12:13}^{A2,1}, \hat{Y}_{13:13}^{A2,1}) = (1.186, 1.369, 1.636)$, respectively.

Table 17.6 The corresponding bias and MSEs for different settings of D approach when the true distribution is ND

D approach							
n	cp	p	$\hat{X}_{s:n}^{A3,1}$		$\hat{X}_{s:n}^{A3,2}$		correct(%)
			bias	MSE	bias	MSE	
20	0.8	0.9	-0.1040	0.0934	-0.2375	0.1343	0.7302
	0.7	0.8	-0.0761	0.0609	-0.1907	0.0890	0.7198
	0.7	0.9	-0.1336	0.1501	-0.1987	0.1661	0.7198
	0.6	0.7	-0.0618	0.0490	-0.1665	0.0725	0.6941
	0.6	0.8	-0.1184	0.1149	-0.1711	0.1259	0.6933
	0.6	0.9	-0.1870	0.2358	-0.2356	0.2480	0.6933
	0.5	0.6	-0.0619	0.0459	-0.1602	0.0664	0.6781
	0.5	0.7	-0.0953	0.0946	-0.1428	0.1036	0.6781
	0.5	0.8	-0.1561	0.1719	-0.1926	0.1801	0.6781
	0.5	0.9	-0.2360	0.3249	-0.2764	0.3358	0.6781
40	0.8	0.9	-0.0335	0.0604	-0.0905	0.0587	0.6921
	0.7	0.8	-0.0168	0.0370	-0.0613	0.0340	0.6924
	0.7	0.9	-0.0364	0.0692	-0.0689	0.0680	0.6924
	0.6	0.7	-0.0100	0.0300	-0.0480	0.0263	0.6998
	0.6	0.8	-0.0162	0.0489	-0.0384	0.0454	0.6990
	0.6	0.9	-0.0278	0.0821	-0.0535	0.0813	0.6990
	0.5	0.6	-0.0084	0.0283	-0.0432	0.0243	0.7536
	0.5	0.7	0.0009	0.0381	-0.0142	0.0337	0.7536
	0.5	0.8	0.0013	0.0548	-0.0132	0.0525	0.7536
	0.5	0.9	-0.0125	0.0832	-0.0334	0.0824	0.7536
60	0.8	0.9	-0.0171	0.0477	-0.0556	0.0425	0.7402
	0.7	0.8	-0.0067	0.0300	-0.0337	0.0249	0.7392
	0.7	0.9	-0.0180	0.0537	-0.0416	0.0505	0.7392
	0.6	0.7	-0.0016	0.0234	-0.0236	0.0188	0.7625
	0.6	0.8	-0.0025	0.0383	-0.0185	0.0344	0.7617
	0.6	0.9	-0.0080	0.0649	-0.0268	0.0629	0.7617
	0.5	0.6	0.0025	0.0226	-0.0182	0.0178	0.7867
	0.5	0.7	0.0173	0.0277	0.0056	0.0237	0.7867
	0.5	0.8	0.0202	0.0400	0.0100	0.0371	0.7867
	0.5	0.9	0.0085	0.0605	-0.0086	0.0587	0.7867

Table 17.7 The corresponding bias and MSEs for different settings of RML approach when the true distribution is SEV

RML approach							
n	cp	p	$\hat{X}_{s:n}^{A1,1}$		$\hat{X}_{s:n}^{A1,2}$		correct(%)
			bias	MSE	bias	MSE	
20	0.8	0.9	0.0034	0.0634	-0.1503	0.0803	0.7460
	0.7	0.8	-0.0090	0.0502	-0.1460	0.0698	0.7011
	0.7	0.9	0.0110	0.1050	-0.0642	0.1038	0.7011
	0.6	0.7	-0.0142	0.0468	-0.1473	0.0662	0.6556
	0.6	0.8	-0.0059	0.0983	-0.0722	0.0982	0.6548
	0.6	0.9	0.0346	0.1863	-0.0212	0.1780	0.6548
	0.5	0.6	-0.0249	0.0527	-0.1570	0.0738	0.6058
	0.5	0.7	-0.0037	0.1006	-0.0655	0.1020	0.6058
	0.5	0.8	0.0258	0.1833	-0.0185	0.1787	0.6058
	0.5	0.9	0.0732	0.3096	0.0272	0.2949	0.6058
40	0.8	0.9	0.1053	0.0562	-0.0159	0.0316	0.9450
	0.7	0.8	0.1001	0.0482	-0.0160	0.0247	0.9401
	0.7	0.9	0.1116	0.0592	0.0296	0.0375	0.9401
	0.6	0.7	0.0939	0.0461	-0.0182	0.0217	0.9354
	0.6	0.8	0.1020	0.0554	0.0220	0.0350	0.9346
	0.6	0.9	0.1229	0.0709	0.0598	0.0523	0.9346
	0.5	0.6	0.0882	0.0486	-0.0222	0.0241	0.9227
	0.5	0.7	0.1044	0.0483	0.0261	0.0299	0.9227
	0.5	0.8	0.1158	0.0565	0.0577	0.0402	0.9227
	0.5	0.9	0.1328	0.0671	0.0825	0.0516	0.9227
60	0.8	0.9	0.0841	0.0392	0.0005	0.0244	0.9817
	0.7	0.8	0.0781	0.0346	-0.0020	0.0189	0.9739
	0.7	0.9	0.0896	0.0403	0.0323	0.0276	0.9739
	0.6	0.7	0.0748	0.0326	-0.0019	0.0168	0.9715
	0.6	0.8	0.0819	0.0385	0.0279	0.0269	0.9707
	0.6	0.9	0.0965	0.0494	0.0531	0.0391	0.9707
	0.5	0.6	0.0664	0.0336	-0.0061	0.0182	0.9633
	0.5	0.7	0.0846	0.0339	0.0339	0.0220	0.9633
	0.5	0.8	0.0957	0.0387	0.0574	0.0301	0.9633
	0.5	0.9	0.1075	0.0460	0.0738	0.0369	0.9633

Example 2

Lieblein and Zelen [22] provided a test on endurance of deep groove ball bearings data and further studied by Meeker and Escobar [23], which is used to illustrate the methodologies developed in this chapter. The data are the numbers of million revolutions before failure for each of the 23 ball bearings in the life test. In the paper of Meeker and Escobar [23], they pointed out that this data (D_2) follows lognormal distribution or Weibull distribution. Hence, $Y_2 = \ln(D_2)$ follows a ND or SEV. The data is given below:

D_2 : 17.88, 28.92, 33.00, 41.52, 42.12, 45.60, 48.40, 51.84, 51.96, 54.12, 55.56, 67.80, 68.64, 68.64, 68.88, 84.12, 93.12, 98.64, 105.12, 105.84, 127.92, 128.04, 173.40.

For more information about this carbon fiber breaking strength data set, one can be referred to Meeker and Escobar [23]. In this example, we assume that the censoring proportion is 0.8696 ($r = 20, n = 23$). Figure 17.3 presents the histogram and the estimated PDFs of ND and SEV based on the type II right-censored data. From Fig. 17.3, it is difficult to decide the best distribution from these two candidate distributions.

We consider using D approach in Example 2 for model selection and use expected values prediction method to predict the future order statistics, which are censored. Based on this censored data, the AMLEs of the model parameters are $(\tilde{\mu}_N, \tilde{\sigma}_N) = (4.147, 0.523)$ and $(\tilde{\mu}_{EV}, \tilde{\sigma}_{EV}) = (4.363, 0.426)$ for the ND and SEV, respectively. The MLEs of μ and σ can be obtained via using Newton-Raphson algorithm with the initial

Table 17.8 The corresponding bias and MSEs for different settings of D_{sp} approach when the true distribution is SEV

D_{sp} approach							
n	cp	p	$\hat{X}_{s:n}^{A2,1}$		$\hat{X}_{s:n}^{A2,2}$		correct(%)
			bias	MSE	bias	MSE	
20	0.8	0.9	0.0344	0.0655	-0.1398	0.0777	0.6417
	0.7	0.8	0.0176	0.0517	-0.1383	0.0677	0.5703
	0.7	0.9	0.0625	0.1159	-0.0295	0.1041	0.5703
	0.6	0.7	0.0141	0.0475	-0.1397	0.0639	0.5019
	0.6	0.8	0.0478	0.1053	-0.0350	0.0958	0.5011
	0.6	0.9	0.1230	0.2154	0.0516	0.1890	0.5011
	0.5	0.6	0.0105	0.0540	-0.1469	0.0708	0.4051
	0.5	0.7	0.0643	0.1107	-0.0163	0.1003	0.4051
	0.5	0.8	0.1271	0.2097	0.0661	0.1871	0.4051
	0.5	0.9	0.2376	0.3965	0.1735	0.3487	0.4051
40	0.8	0.9	0.1253	0.0555	-0.0073	0.0324	0.9077
	0.7	0.8	0.1193	0.0455	-0.0089	0.0248	0.8803
	0.7	0.9	0.1380	0.0630	0.0459	0.0421	0.8803
	0.6	0.7	0.1115	0.0421	-0.0121	0.0217	0.8590
	0.6	0.8	0.1254	0.0546	0.0347	0.0372	0.8582
	0.6	0.9	0.1522	0.0804	0.0812	0.0617	0.8582
	0.5	0.6	0.1067	0.0438	-0.0163	0.0239	0.8438
	0.5	0.7	0.1234	0.0463	0.0352	0.0309	0.8438
	0.5	0.8	0.1421	0.0597	0.0749	0.0449	0.8438
	0.5	0.9	0.1633	0.0799	0.1072	0.0640	0.8438
60	0.8	0.9	0.0997	0.0409	0.0069	0.0254	0.9500
	0.7	0.8	0.0936	0.0344	0.0033	0.0194	0.9280
	0.7	0.9	0.1125	0.0458	0.0460	0.0317	0.9280
	0.6	0.7	0.0906	0.0312	0.0028	0.0170	0.9124
	0.6	0.8	0.1036	0.0404	0.0390	0.0289	0.9116
	0.6	0.9	0.1230	0.0582	0.0721	0.0464	0.9116
	0.5	0.6	0.0847	0.0319	-0.0014	0.0183	0.8991
	0.5	0.7	0.1042	0.0345	0.0425	0.0232	0.8991
	0.5	0.8	0.1197	0.0438	0.0724	0.0342	0.8991
	0.5	0.9	0.1382	0.0460	0.0977	0.0470	0.8991

Table 17.9 The corresponding bias and MSE for different settings of D approach when the true distribution is SEV

D approach							
n	cp	p	$\hat{X}_{s:n}^{A3,1}$		$\hat{X}_{s:n}^{A3,2}$		correct(%)
			bias	MSE	bias	MSE	
20	0.8	0.9	0.0335	0.0654	-0.1401	0.0778	0.6450
	0.7	0.8	0.0173	0.0516	-0.1384	0.0677	0.5723
	0.7	0.9	0.0621	0.1156	-0.0297	0.1040	0.5723
	0.6	0.7	0.0141	0.0475	-0.1398	0.0639	0.5022
	0.6	0.8	0.0478	0.1054	-0.0350	0.0958	0.5014
	0.6	0.9	0.1230	0.2154	0.0515	0.1890	0.5014
	0.5	0.6	0.0105	0.0540	-0.1469	0.0708	0.4050
	0.5	0.7	0.0644	0.1107	-0.0163	0.1003	0.4050
	0.5	0.8	0.1271	0.2097	0.0661	0.1871	0.4050
	0.5	0.9	0.2377	0.3965	0.1736	0.3487	0.4050
40	0.8	0.9	0.1226	0.0544	-0.0083	0.0322	0.9135
	0.7	0.8	0.1173	0.0454	-0.0095	0.0248	0.8839
	0.7	0.9	0.1366	0.0625	0.0451	0.0419	0.8839
	0.6	0.7	0.1109	0.0420	-0.0122	0.0217	0.8606
	0.6	0.8	0.1249	0.0545	0.0345	0.0371	0.8598
	0.6	0.9	0.1513	0.0800	0.0807	0.0615	0.8598
	0.5	0.6	0.1068	0.0438	-0.0162	0.0239	0.8433
	0.5	0.7	0.1235	0.0463	0.0352	0.0309	0.8433
	0.5	0.8	0.1422	0.0598	0.0749	0.0449	0.8433
	0.5	0.9	0.1634	0.0799	0.1073	0.0641	0.8433
60	0.8	0.9	0.0968	0.0400	0.0058	0.0252	0.9563
	0.7	0.8	0.0926	0.0341	0.0030	0.0193	0.9309
	0.7	0.9	0.1114	0.0452	0.0454	0.0315	0.9309
	0.6	0.7	0.0902	0.0311	0.0027	0.0170	0.9135
	0.6	0.8	0.1029	0.0402	0.0387	0.0288	0.9127
	0.6	0.9	0.1219	0.0580	0.0715	0.0463	0.9127
	0.5	0.6	0.0847	0.0319	-0.0014	0.0183	0.8990
	0.5	0.7	0.1043	0.0345	0.0426	0.0232	0.8990
	0.5	0.8	0.1198	0.0438	0.0724	0.0342	0.8990
	0.5	0.9	0.1383	0.0572	0.0978	0.0470	0.8990

values of $(\tilde{\mu}_N, \tilde{\sigma}_N)$ and $(\tilde{\mu}_{EV}, \tilde{\sigma}_{EV})$; the resulting MLEs are $(\hat{\mu}_N, \hat{\sigma}_N) = (4.148, 0.524)$ and $(\hat{\mu}_S, \hat{\sigma}_S) = (4.369, 0.425)$ for the ND and SEV, respectively. The D values based on using the ND and SEV are 0.181 and 0.297, respectively. Because the D value obtained from ND is smaller than that obtained from SEV, we claim the best model is normal. The expected values prediction of $(Y_{21:23}, Y_{22:23}, Y_{23:23})$ via using ND is $(\hat{Y}_{21:23}^{A3,1}, \hat{Y}_{22:23}^{A3,1}, \hat{Y}_{23:23}^{A3,1}) = (4.784, 4.922, 5.160)$. In addition, we compare our prediction results with the MMLP values that proposed by Yang and Tong [8]. The MMLP of $(Y_{21:23}, Y_{22:23}, Y_{23:23})$ is $(\hat{Y}_{21:23}, \hat{Y}_{22:23}, \hat{Y}_{23:23}) = (4.662, 4.936, 5.175)$. Our predicted results are close to that proposed by Yang and Tong [8] even we cannot initially assume which one of the ND or SEV is the best distribution.

Example 3

A survival time (in days) of pigs injected with different doses of tubercle Bacilli dataset was provided in Bjerkedal [24]. Because guinea pigs have a high susceptibility to human tuberculosis, they are often used for research. There were 72 observations listed below:

- $D_3 : 12, 15, 22, 24, 24, 32, 32, 33, 34, 38, 38, 43, 44, 48, 52, 53,$
- $54, 54, 55, 56, 57, 58, 58, 59, 60, 60, 60, 60, 61, 62, 63, 65,$
- $65, 67, 68, 70, 70, 72, 73, 75, 76, 76, 81, 83, 84, 85, 87, 91, 95,$
- $96, 98, 99, 109, 110, 121, 127, 129, 131, 143, 146, 146, 175,$
- $175, 211, 233, 258, 258, 263, 297, 341, 341, 376.$

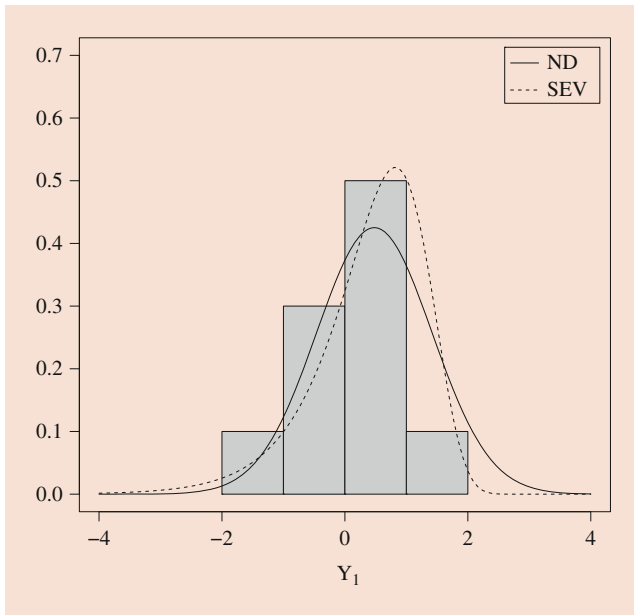


Fig. 17.2 The histogram and the estimated PDFs of airplane component's failure time in Example 1

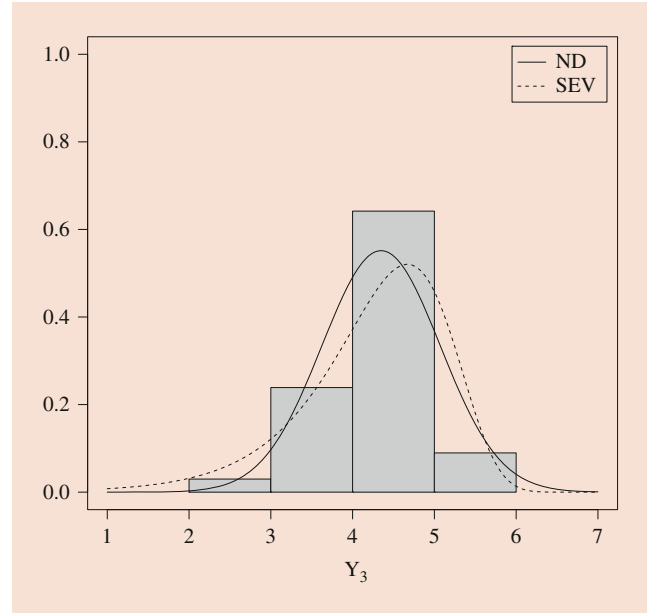


Fig. 17.4 The histogram and the estimated PDFs of survival times (in days) of pigs injected in Example 3

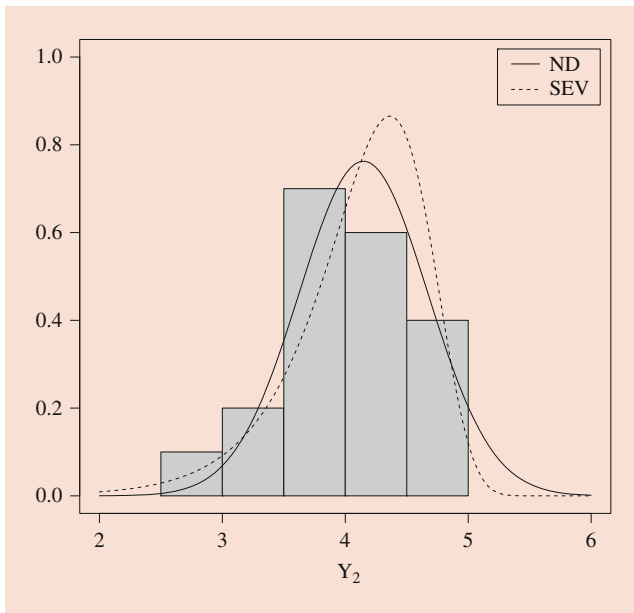


Fig. 17.3 The histogram and the estimated PDFs of tests on endurance of deep groove ball bearings in Example 2

In this example, we assume that the censoring proportion is 0.930 ($r = 67, n = 72$). Let Y_3 be the logs of the ten observations, that is, $Y_3 = \ln(D_3)$.

Figure 17.4 presents the histogram and the estimated PDFs of the ND and SEV of Y_3 . From Fig. 17.4, we find that it is difficult to decide the best distribution from the ND

or SEV distribution. We consider using RML approach for model selection and use Taylor series prediction method to predict the future order statistics, which are censored.

Based on this censored data, the AMLEs of the model parameters in ND and SEV distribution are obtained by $(\tilde{\mu}_N, \tilde{\sigma}_N) = (4.348, 0.723)$ and $(\tilde{\mu}_{EV}, \tilde{\sigma}_{EV}) = (4.671, 0.707)$, respectively. The MLEs of μ and σ are then obtained via using the Newton-Raphson algorithm with the initial values of $(\tilde{\mu}_N, \tilde{\sigma}_N)$ and $(\tilde{\mu}_{EV}, \tilde{\sigma}_{EV})$, and the resulting MLEs are $(\hat{\mu}_N, \hat{\sigma}_N) = (4.348, 0.723)$ and $(\hat{\mu}_{EV}, \hat{\sigma}_{EV}) = (4.671, 0.707)$ for the ND and SEV distribution, respectively. The value of τ is evaluated with $\tau = 7.302$. Because $\hat{L}_N(\hat{\mu}_N, \hat{\sigma}_N) > \hat{L}_S(\hat{\mu}_S, \hat{\sigma}_S)$, we claim that this data is automatically determined to have a ND. The Taylor series prediction of $(Y_{68:72}, Y_{69:72}, Y_{70:72}, Y_{71:72}, Y_{72:72})$ for the ND is $(\hat{Y}_{68:72}^{A1,2}, \hat{Y}_{69:72}^{A1,2}, \hat{Y}_{70:72}^{A1,2}, \hat{Y}_{71:72}^{A1,2}, \hat{Y}_{72:72}^{A1,2}) = (5.553, 5.553, 5.626, 5.711, 5.836)$.

17.6 Conclusions

In the real world, it is possible that we cannot make a clear judgment of the distribution of data, especially in the case of censored data. In this study, we focus on providing reliable methods to obtain predicting values of censored data to reduce the impact of model misspecification. We provide two prediction methods and three model selection approaches for type II censored data, in which the quality characteristic is

assumed to follow a location-scale family. The ND and SEV are considered as the candidate members in the location-scale distribution to compete for the best underlying distribution.

Numerical results show that the three proposed model selection approaches are robust and effective for obtaining good predicted values for the future order statistics, which are censored. In comparing these three proposed approaches, we recommend using D_{SP} approach or D approach for model selection and use expected values prediction method to predict the future order statistics for n is less than 40. When n is larger than 40, we recommend using RRML approach for model selection and using Taylor series prediction method to predict the future order statistics. The simulation results also show that the proposed approaches have strong robustness and can greatly reduce the impact of model assumption uncertainty. If more than two candidate distributions are competing for the optimal distribution, the proposed method can also work well.

In future researches, it is worthwhile to improve the accuracy of model selection or to predict other types of censored data. Other model selection methods from the current three proposed approaches could also be competitive. How to apply the new model selection method for the topic of type II censored data prediction could be studied in the future.

References

- Kaminsky, K.S., Nelson, P.I.: Prediction of order statistics. In: Balakrishnan, N., Rao, C.R. (eds.) *Handbook of Statistics 17, Order Statistics: Applications*, pp. 431–450. Elsevier, New York (1998)
- Fertig, K.W., Meyer, M.E., Mann, N.R.: On constructing prediction intervals for samples from a Weibull or extreme value distribution. *Technometrics*. **22**, 567–573 (1980)
- Kaminsky, K.S., Rhodin, L.S.: Maximum likelihood prediction. *Ann. Inst. Stat. Math.* **37**, 507–517 (1985)
- Wu, J.-W., Lu, H.-L., Chen, C.-H., Yang, C.-H.: A note on the prediction intervals for a future ordered observation from a Pareto distribution. *Qual. Quant.* **38**, 217–233 (2004)
- Kundu, D., Raqab, M.Z.: Bayesian inference and prediction of order statistics for a Type-II censored Weibull distribution. *J. Stat. Plan. Inference*. **142**(1), 41–47 (2012)
- Panahi, H., Sayyareh, A.: Parameter estimation and prediction of order statistics for the Burr type XII distribution with type II censoring. *J. Appl. Stat.* **41**(1), 215–232 (2014)
- Raqab, M.Z.: Modified maximum likelihood predictors of future order statistics from normal samples. *Comput. Stat. Data Anal.* **25**(1), 91–106 (1997)
- Yang, C.H., Tong, L.I.: Predicting type II censored data from factorial experiments using modified maximum likelihood predictor. *Int. J. Adv. Manuf. Technol.* **30**, 887–896 (2006)
- Chiang, J.-Y.: Modified maximum likelihood prediction for type II censored data under the Weibull distribution. *Int. J. Intell. Technol. Appl. Stat.* **3**(1), 17–32 (2010)
- Dumoncaux, R., Antle, C.E.: Discriminating between the log-normal and Weibull distribution. *Technometrics*. **15**, 923–926 (1973)
- Kundu, D.K., Manglick, A.: Discriminating between the log-normal and gamma distributions. *J. Appl. Stat. Sci.* **14**, 175–187 (2005)
- Kundu, D., Raqab, M.Z.: Discriminating between the generalized Rayleigh and log-normal distribution. *Statistics*. **41**(6), 505–515 (2007)
- Yu, H.F.: Mis-specification analysis between normal and extreme value distributions for a screening experiment. *Comput. Ind. Eng.* **56**(4), 1657–1667 (2009)
- Ashour, S.K., Hashish, A.M.: A numerical comparison of three procedures used in failure model discrimination. *Pak. J. Stat. Oper. Res.* **10**(1), 107–119 (2014)
- Pakyari, R.: Discriminating between generalized exponential, geometric extreme exponential and Weibull distributions. *J. Stat. Comput. Simul.* **80**(12), 1403–1412 (2010)
- Elsherpieny, A.E., Muhammed, Z.H., Radwan, U.N.: On discriminating between gamma and log-logistic distributions in case of progressive type II censoring. *Pak. J. Stat. Oper. Res.* **13**(1), 157–183 (2017)
- Chiang, J.-Y., Wang, S., Tsai, T.-R., Li, T.: Model selection approaches for predicting future order statistics from type II censored data. *Math. Probl. Eng.*, 3465909 (2018). <https://doi.org/10.1155/2018/3465909>
- Hossain, A., Willan, A.R.: Approximate MLEs of the parameters of location-scale models under type II censoring. *Statistics*. **41**(5), 385–394 (2007)
- Teichroew, D.: Tables of expected values of order statistics and products of order statistics for samples of size twenty and less from the normal distribution. *Ann. Math. Stat.* **27**, 410–426 (1956)
- Castro-Kuriss, C., Kelmansky, D.M., Leiva, V., Martizez, E.J.: A new goodness-of-fit test for censored data with an application in monitoring process. *Commun. Stat. Simul. Comput.* **38**(6), 1161–1177 (2009)
- Mann, N.R., Fertig, K.W.: Tables for obtaining confidence bounds and tolerance bounds based on best linear invariant estimates of parameters of the extreme value distribution. *Technometrics*. **15**, 87–101 (1973)
- Lieblein, J., Zelen, M.: Statistical investigation of the fatigue life of deep-groove ball bearings. *J. Res. Natl. Bur. Stand.* **57**, 273–316 (1956)
- Meeker, W.Q., Escobar, L.A.: *Statistical Methods for Reliability Data*. Wiley, New York (1998)
- Bjerkedal, T.: Acquisition of resistance in guinea pigs infected with different doses of virulent tubercle bacilli. *Am. J. Hyg.* **72**, 130–148 (1960)



Tzong-Ru Tsai is the Dean of the College of Business and Management. He is also a Professor in the Department of Statistics at Tamkang University, Taiwan. His main research interests include quality control and reliability analysis and he has served as a consultant for many companies. He is an associate editor of the *Journal of Statistical Computation and Simulation*.



Jun-You Chiang is an assistant professor in the School of Statistics, Southwestern University of Finance and Economics. He received his PhD in management from the Graduate Institute of Management Sciences at Tamkang University. His research interests include reliability, quality control, and applied statistics.



Yan Qin is a lecturer in the School of Foreign Languages for Business, Southwestern University of Finance and Economics. Her main research interests include economic model analysis and applied statistics.



Shuai Wang is a doctoral student in the School of Statistics, Southwestern University of Finance and Economics. His main research interests include quality control, model selection, and econometrics.



Monitoring Coefficient of Variation Using CUSUM Control Charts

Phuong Hanh Tran , Huu Du Nguyen , Cédric Heuchenne , and Kim Phuc Tran

Contents

18.1	The CUSUM Control Charts	335
18.1.1	The Cumulative Sum.....	335
18.1.2	The CUSUM Statistics.....	336
18.1.3	The Implementation of a CUSUM Control Chart.....	337
18.1.4	A Measure to Evaluate the Performance of a CUSUM Control Chart.....	337
18.2	CUSUM Control Chart Monitoring the CV	339
18.2.1	Basic Properties of Sample CV Squared.....	339
18.2.2	Implementation and Design of CUSUM- γ^2 Control Charts.....	340
18.2.3	Performance of the CUSUM- γ^2 Control Charts.....	341
18.3	CUSUM Control Chart Monitoring the CV in the Presence of Measurement Error	342
18.3.1	Linear Covariate Error Model for the Coefficient of Variation.....	344
18.3.2	The Effect of Measurement Error on the CUSUM- γ^2 Control Charts.....	345
18.4	Using FIR to Improve the Performance of the CUSUM-γ^2 Control Chart	353

18.5	Conclusions and Perspectives	357
-------------	-------------------------------------------	-----

References	357
-------------------------	-----

Abstract

In the field of statistical process control, the cumulative sum (CUSUM) control chart is used as a powerful tool to detect process shifts. One of the main features of the CUSUM control chart is that it takes into account the past information at each sampling time of the process. Recently, the rapid development of optimization algorithms and software makes the CUSUM chart easier to be implemented. As a result, the CUSUM control chart has been increasingly applied widely. The goal of this chapter is to present some recent innovative CUSUM control charts monitoring the coefficient of variation (CV). We address several problems related to the CUSUM chart monitoring the CV. The first section provides important definitions of a CUSUM control chart, including the CUSUM sequence, the CUSUM statistics, the implementation of a CUSUM control chart, the average run length (ARL), and the expected average run length (EARL). In the second section, we investigate the effect of measurement error on the CUSUM control chart monitoring the CV. Finally, a fast initial response strategy to improve the performance of the CUSUM control chart is presented.

Keywords

Cumulative sum · Coefficient of variation · Measurement error · Fast initial response · Statistic process control · Control chart

Control charts are very powerful tools widely used in statistical process control (SPC) to detect assignable causes that lead to changes in the process output. Ever since being introduced by Shewhart, it has become a common practice for

The original version of this chapter was revised. Correction to this chapter can be found at https://doi.org/10.1007/978-1-4471-7503-2_57

P. H. Tran (✉)
HEC-Liège, Management School of the University of Liège, Liège, Belgium

University of Lille, ENSAIT, ULR 2461 - GEMTEX - Genie et Matériaux Textiles, Lille, France
e-mail: phtran@uliege.be

H. D. Nguyen (✉)
International Research Institute for Artificial Intelligence and Data Science, Dong A University, Danang, Vietnam
e-mail: dunh@donga.edu.vn

C. Heuchenne (✉)
HEC-Liège, Management School of the University of Liège, Liège, Belgium
e-mail: C.Heuchenne@ulg.ac.be

K. P. Tran (✉)
University of Lille, ENSAIT, ULR 2461 - GEMTEX - Génie et Matériaux Textiles, Lille, France
e-mail: kim-phuc.tran@ensait.fr

practitioners to use various control charts to monitor different processes. The practical applications of control charts, which were primarily in manufacturing, now extend far beyond manufacturing into a large number of other fields in real life, such as engineering, environmental science, genetics, epidemiology, medicine, finance, and even law enforcement and athletics (Stoumbos and Reynolds [1]).

A standard Shewhart control chart consists of a center line and the control limits (upper and lower). A process is monitored using this control chart by plotting directly the statistical values of interest on the chart. If these values exceed the control limits, the chart will signal an alarm to show that the process is in out-of-control condition, and a special case could occur that affects the process. Otherwise, the process is considered to be still in control.

It is well documented that in the SPC literature, the traditional Shewhart control chart is quite slow in detecting small or moderate shifts. The main reason is that this type of control chart is based purely on the latest update of observations from the process instead of utilizing the information embedded in the entire sequence of measurements. In other words, it has no “memory.” To overcome this drawback, other advanced control charts with “memory” have been suggested such as the exponentially weighted moving average (EWMA) control chart and the cumulative sum (CUSUM) control chart. During the last decade, the CUSUM control chart has been increasingly used as a powerful tool in SPC to detect small shifts in the process of interest. The CUSUM control chart takes into account the past information at each sampling time of the process. As a result, it is very sensitive to small process shifts. In recent years, the rapid development of optimization algorithms and software makes the CUSUM chart easier to be implemented. Thus, similar to the discussion about the application of the EWMA chart in Castagliola et al. [2], the CUSUM chart could be applied in various industrial environments, such as (1) the semiconductor industry at the level of wafer fabrication where the extremely high level of precision in critical dimensions of parts is required to avoid the rejection of the product at the testing stage or during the operating conditions, (2) automotive manufacturing industry where the technological process of producing mechanical parts like CNC operations in machining centers requires a small variability as possible, and (3) service control activities where healthcare outcomes such as the occurrence of infections are monitored. In general, the CUSUM chart is suitable for any process or service when it is expected to detect small process shifts in the process. In the SPC literature, the CUSUM control chart has also been a major concern in several studies. Chang and McLean [3] used the CUSUM chart as an early performance indicator of a clinical procedure before its implementation. Yashchin [4] studied CUSUM control schemes for serially correlated observations with an

application to the process of galvanic plating used to deposit copper on the surface of certain items. A CUSUM-based method for monitoring simple linear profiles and the variance when parameters are estimated was introduced by Saghaei et al. [5] and Castagliola and Maravelakis [6], respectively. Hawkins and Olwell [7] provided an extensive overview of the CUSUM methodology for quality improvement in SPC.

The coefficient of variation (CV) is a quantity to evaluate the dispersion of a probability distribution. It is defined as the ratio of the standard deviation to the mean of the distribution. In many practical situations, this measure plays an important role in ensuring the quality and stability of the process. More specifically, the mean of the quantity of interest is expected to vary from time to time and its standard deviation changes with the mean. However, the process is still considered to be in control as long as the CV remains stable. Many opportunities for control chart monitoring the sample CV have been discussed in industrial manufacturing. In the field of materials engineering like tool cutting life and sintered materials, some quality characteristics related to the physical properties of products constituted by metal alloys or composite materials often have a standard deviation which is proportional to their population mean due to the way atoms of a metal diffuse into another (Castagliola et al. [8]). In the textile industry, the variation among tensile strength measurements from the thin thread is significantly smaller than that from heavy thread because of the inherent physical properties of fiber (Yeong et al. [9]). In the machining process, monitoring the CV is useful to detect the presence of chatter, a severe form of self-excited vibration which results in many machining problems: by adopting the CV, the process monitoring works for different machining materials and machining parameters (Ye et al. [10]). More applications of monitoring of the CV have been mentioned by Muhammad et al. [11].

On account of the importance of monitoring the CV in many processes, a large number of the CV control charts has been suggested in the literature. Through a clinical chemical example, Kang et al. [12] were pioneering authors to show the efficiency of the Shewhart CV chart in quality improvement, when neither the process mean nor the process variance is constant. However, this control chart is only effective in detecting large CV shifts. For small or moderate CV shifts, it takes a large number of samples plotted beyond the control interval before the Shewhart CV chart signals an alarm. The EWMA CV control chart was then studied by Hong et al. [13] to improve the performance of the Shewhart CV chart. Using the same EWMA control chart, Castagliola et al. [8] pointed out that monitoring the CV squared is more effective than monitoring the CV itself. The performance of the standard Shewhart CV control chart could also be improved by the Shewhart chart with supplementary run

rules (Castagliola et al. [14]). Recently, Tran et al. [15] improved the performance of run rules control chart [14] by using one-sided run rules to monitor CV squared. They also considered the performance of these charts in the presence of measurement errors (ME). Moreover, another strategy that could be used to improve the performance of Shewhart chart is the synthetic CV chart (Calzada and Scariano [16]). A comparison between the performances of synthetic and EWMA charts for monitoring the CV made by Teoh et al. [17] showed that the EWMA CV squared chart outperforms the synthetic CV chart for detecting small shifts in the CV. Based on the preliminary work of Castagliola et al. [8], Zhang et al. [18] presented a modified EWMA CV control chart that is superior to some other competing control charts. Furthermore, Hong et al. [19] developed a CV control chart using a generally weighted moving average (GWMA) technique, called the GWMA CV control chart. The authors showed that this new type of control chart brings more excellent performance than the EWMA control chart in detecting small shifts in the CV. Subsequently, the run sum CV control chart was investigated by Teoh et al. [20]. Tran and Tran [21] considered two one-sided CUSUM CV control charts and their control charts lead to better performance than some previous ones. In addition, the adaptive strategies were also adopted to develop new charts monitoring the CV. Recently, Tran et al. [22] suggested CUSUM control charts to monitor CV squared with a fast initial response (FIR) strategy. The performance of FIR CUSUM CV squared charts in their results is greater than the initial CUSUM CV squared ones. Furthermore, the variable sampling interval (VSI) CV control chart was first proposed by Castagliola et al. on 2013. [23]. Although its performance cannot overcome the performance of other advanced charts like EWMA CV chart or CUSUM CV chart, this kind of chart is still a major concern in several situations because of its simplicity in implementation and its better performance compared to the standard Shewhart CV chart. Then, the variable sample size strategy was adopted for the CV control chart by Castagliola et al. [24] and Yeong et al. [25] where the first author group monitored a transformed statistic of the CV while the second one monitored directly the CV without any transformation. Subsequently, the variable sampling interval and variable sample size (VSSI) CV control charts were investigated by Khaw et al. [26]. At the highest level of adaptive strategies, Yeong et al. [27] designed the variable parameter (VP) chart that includes all VSSI, VSI, and VSS strategies to monitor the CV. It was shown in this study that the VP CV chart outperforms the VSSI CV, VSI CV, VSS CV, and Shewhart CV charts in detecting the CV shifts. The adaptive strategies were also combined efficiently with the advanced control chart. For example, the VSI EWMA CV control chart and the VSS EWMA CV control chart were proposed by Yeong et al. [28] and Muhammad et al. [11] with higher performance than the

original EWMA CV control chart, respectively. Lately, Tran et al. [29] investigated on VSI CUSUM control charts that have highest performance in comparison with previous charts such as, respectively, original CUSUM, EWMA, and VSI EWMA control charts. Other problems related to the design of a control chart monitoring the CV were also considered in the SPC literature. For instance, the CV control charts in short production runs were investigated by Castagliola et al. [30] with a Shewhart chart and Amdouni et al. [31–33] with a VSI chart, a run rules chart, and a VSS chart, respectively. The performance of the CV control charts in Phase I data was considered by Dawod et al. [34]. The effect of ME on the CV control chart is investigated by Yeong et al. [35] with the Shewhart type chart, Tran et al. [36] with the synthetic chart, Nguyen et al. [37] with the VSI type chart, and Tran et al. [38] with the CUSUM-type chart respectively.

Table 18.1 provides a brief overview of control chart monitoring the CV in the literature.

The goal of this chapter is to present some recent innovative CUSUM control charts monitoring the CV. A number of important problems related to the CUSUM CV control chart will be addressed. First, we describe several definitions used in the design of a CUSUM control chart, including the definition of a CUSUM sequence, the CUSUM statistics, the implementation of a CUSUM control chart, and the widely used measures to evaluate the performance of a CUSUM control chart, i.e., the average run length (ARL) and the expected average run length (EARL). This is to provide practitioners with a quick overview and a comprehensive understanding of a CUSUM control chart. Then, the effect of ME on the CUSUM control chart monitoring the CV is investigated using a linear covariate error model. The obtained results from this section are important for applying the CUSUM chart in reality because it is necessary to understand the effect of the ME, which is unavoidable in practice, and discuss the method to reduce the negative effects on the CUSUM CV chart performance. In the next section, we present a fast initial response strategy to improve the performance of the CUSUM control chart. Some perspectives related to designing a control chart monitoring the CV are discussed in the conclusion.

18.1 The CUSUM Control Charts

18.1.1 The Cumulative Sum

Let Y_1, Y_2, \dots be a sequence of independent observations, where Y_i represents the measurement of a quality of interest at time i . Let μ_0 denote the target value. From the sequence Y_1, Y_2, \dots , one can define a new sequence, say the cumulative sum sequence, as

Table 18.1 The CV control charts in the SPC literature

Type of control chart	References	Additional description
Shewhart chart	Kang et al. [12] Castagliola et al. [30] Dawod et al. [34] Yeong et al. [35]	Short production runs Phase I data Considering ME
Synthetic chart	Calzada and Scariano [16]	
EWMA chart	Hong et al. [13] Castagliola et al. [8] Zhang et al. [18]	Monitoring the CV squared Modified EWMA chart
GWMA chart	Hong et al. [19]	
CUSUM chart	Tran and Tran [21] Tran et al. [38] Tran et al. [22]	Considering ME Considering FIR strategy
Run rules chart	Castagliola et al. [14] Amdouni et al. [32] Tran et al. [15]	One-sided run rules in short production runs One-sided run rules; monitoring the CV squared; considering ME
Run sum chart	Teoh et al. [20]	
VSI chart	Castagliola et al. [23] Amdouni et al. [31] Nguyen et al. [37]	Short production runs Considering ME
VSS chart	Yeong et al. [25] Castagliola et al. [24] Amdouni et al. [33]	Monitoring a transformed statistic of the CV Short production runs
VSSI chart	Khaw et al. [26]	
VP chart	Yeong et al. [27]	
VSI EWMA chart	Yeong et al. [28]	
VSS EWMA chart	Muhammad et al. [11]	
VSI CUSUM chart	Tran et Heuchenne [29]	Monitoring the CV squared

$$C_i = \sum_{j=1}^i (Y_j - \mu_0), \quad i = 1, 2, \dots$$

The new sequence works by accumulating the deviations from μ_0 . Remark that this sequence can be rewritten in a recursive form as

$$C_0 = 0,$$

$$C_i = C_{i-1} + (Y_i - \mu_0), \quad i = 1, 2, \dots$$

This form explains the memory of the actual CUSUM: its mechanism is to add the current value $(Y_i - \mu_0)$ to the previous memory C_{i-1} . The effect of the mean shift on C_i can be seen by the formula

$$E(C_i) = E(C_{i-1}) + \mu_i - \mu_0,$$

where $\mu_i = E(Y_i)$, $i = 1, 2, \dots$. Indeed, if $\mu_i = \mu_0$, then $E(C_i) = E(C_{i-1})$ and the cumulative sum C_i remains where it was. If $\mu_i > \mu_0$, then $E(C_i) > E(C_{i-1})$ and C_i tends to be shifted upward. By contrast, when $\mu_i < \mu_0$, C_i tends to be

shifted downward since $E(C_i) < E(C_{i-1})$. That is to say, the trend in C_i is an indicator of changes in the process.

18.1.2 The CUSUM Statistics

In a practical implementation of the CUSUM control chart, the cumulative sum C_i is not monitored directly. Instead, the following is the statistics of interest:

$$C_i^+ = \max[0, C_{i-1}^+ + Y_i - (\mu_0 + K^+)], \quad (18.1)$$

$$C_i^- = \max[0, C_{i-1}^- + (\mu_0 - K^-) - Y_i], \quad (18.2)$$

where the starting values are $C_0^+ = C_0^- = 0$ and the parameters $K^+(K^-)$ are predetermined positive constants.

Compared to the original CUSUM statistic, the new formulation has more advantages. First, the simultaneous use of two separated statistics C_i^+ and C_i^- , which are called one-sided upper and lower CUSUM, allows decoupling of the positive and negative mean shifts so that they can eliminate the previous effect of each other. Second, the constant $K^+(K^-)$ plays the role of a "safeguarding threshold" to prevent small fluctuations from the accumulation and to cause

the CUSUM chart to signal. In the literature, K^+ (K^-) is called the reference value (or the allowance). For simplicity of presentation, sometimes we denote K as a general reference value for both cases.

18.1.3 The Implementation of a CUSUM Control Chart

Like other control charts, it is necessary to assign a predetermined threshold for the one-sided CUSUM control charts, called the control limit. This threshold is denoted by H^+ or UCL for the upper control chart (the control chart that aims to detect the increasing shifts) and H^- or LCL for the lower one (the control chart that aims to detect the decreasing shifts). Especially, when we do not want to emphasize the one-sided charts, we denote this value by H for a general case. Then, in the implementation of the one-sided CUSUM control chart, the statistics C_i^+ and C_i^- are plotted on it. If either C_i^+ or C_i^- exceeds the control limits, i.e., over H^+ for the upper chart and under H^- for the lower chart, the process is considered to be out-of-control. In contrast, when these statistics are in the interval control, i.e., under H^+ for the upper chart and above H^- for the lower chart, the process is said to be still in-control.

The CUSUM control chart is designed by assigning values for the reference parameter K and the control limit H . There are different advices for choosing K and H depending on the situation under study. For example, in the CUSUM control chart monitoring the process mean, it is suggested to choose K equal to one-half the magnitude of the shift size and H equal to five times the process standard deviation (Douglas [39]). In general cases, these values are usually chosen based on some constraints on the standard measure that is used to evaluate the performance of the control chart. We will present a widely used measure called the average run length (ARL) in the next section. In the design of the CUSUM control charts, the parameters K and H are selected to satisfy two following constraints: (1) the in-control ARL_0 value is equal to a predetermined desired value, and (2) the out-of-control ARL_1 value is minimized.

18.1.4 A Measure to Evaluate the Performance of a CUSUM Control Chart

The Average Run Length

At the beginning of the implementation, the initial states C_0^+ and C_0^- are plotted on the axis (as in the general case, we suppose $C_0^+ = C_0^- = 0$). Then, the change in the process is cumulative and the statistics C_i^+ and C_i^- are added by positive values. The number of points (or samples) from the starting

point up to the point at which the control limit is crossed is called the run length (RL). This is a random variable taking integer values only, $RL \in \{1, 2, 3, \dots\}$.

There is an analogy between a false alarm of a control chart and a Type I in classical hypothesis testing. A false alarm means that the CUSUM signals when there is no shift occurred. It makes the process be interrupted and waste time when one would like to find out nonexistent special causes. Therefore, the runs between unavoidable false alarms are expected to be as long as possible. Similarly, a Type II error in classical hypothesis testing is analogous to the situation where the CUSUM is still within its control limits even though the process has been shifted because of assignable causes. To attain the quality of the process, whenever an assignable cause occurs that shifts the process, we would like to detect them as soon as possible. That is to say, it is expected in practice to have a short run before the chart signals actual shifts. The expectation of long runs before false alarms conflicts with that of short runs before the control chart signals an actual shift. In practice, a trade-off between these two objectives is made by using the average run length (ARL), which is the mean of RL .

By its definition, the ARL represents the expected number of samples on which a control chart first signals, i.e., the number of samples inside the control limits before the first point indicates an out-of-control condition. When the process is in-control, it is normally denoted by ARL_0 ; otherwise, it is denoted by ARL_1 . The trade-off between two conflicting desires is now implemented by fixing the ARL_0 at an acceptable value and minimizing the ARL_1 .

As discussed by Douglas and David [40], the ARL is not a perfect measure for the general tendency toward long or short runs. Since the run length distribution could be highly variable, a high ARL_0 does not always lead to a very short run before a false alarm given by the chart. Similarly, a low ARL_1 does not make sure that an actual shift could be detected after a not long run. However, the ARL measure has its own advantage in that it is easy to be interpreted and well-defined. Therefore, it is considered as a standard measure of performance of the CUSUM control chart. When the CUSUM control chart is combined with other adaptive strategies like variable sampling interval and variable sample size, this measure is transformed into other measures like average time to signal (ATS, the expected length of time from the start of process monitoring until a signal is generated), the average number of observations to signal (ANOS, the expected number of observations from the start of process monitoring until a signal is given), and the average number of samples to signal (ANSS, the expected number of observations and the expected number of samples from the start of process monitoring until a signal is given). However, the calculation of these measures is based on the calculation of the ARL .

The widely used methods to calculate the *ARL* are given in the sequel.

Methods to Calculate the *ARL*

In the literature, several methods to calculate the *ARL* for a CUSUM control chart are available. For the CUSUM control chart monitoring the process mean, one can use Siegmund’s approximation as suggested by Woodall and Adams [41] or Hawkins’s approximation as presented by Hawkins [42]. In general cases, one can use the Fredholm integral equation proposed by Page [43] and Crowder [44] or the Markov chain approach proposed by Brook and Evans [45]. Among these methods, the Markov chain approach and the Fredholm integral equation are widely used due to their flexibility and simplicity.

The Markov Chain Method

The states of the Markov chain are defined by partitioning the control interval into a finite set of subintervals and by considering the midpoint value within each subinterval to approximate the value of the CUSUM statistic. In particular, we assume to have a discrete-time Markov with $p + 1$ different transitional states, where states $0, 1, \dots, p - 1$ are transient states and state p is an absorbing one corresponding to a signal from the CUSUM control chart. This is equivalent to dividing the control interval into p subintervals; each subinterval represents a transient state of a Markov chain. To get a better approximation, the “restart state” should be represented by the first subinterval which has half size of others. Figure 18.1 demonstrates this subdivision for the upward chart, where the threshold is denoted by H^+ . The transition probability matrix \mathbf{P} of this discrete-time Markov chain is

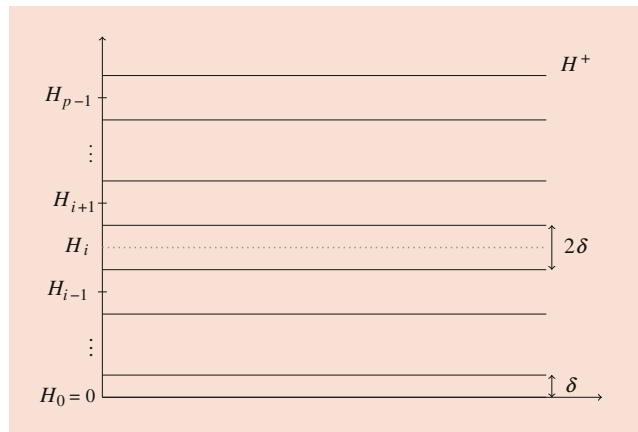


Fig. 18.1 A subdivision of the control limit interval of upward chart

$$\mathbf{P} = \begin{pmatrix} \mathbf{Q} & \mathbf{r} \\ \mathbf{0}^T & 1 \end{pmatrix} = \begin{pmatrix} Q_{0,0} & Q_{0,1} & \dots & Q_{0,p-1} & r_0 \\ Q_{1,0} & Q_{1,1} & \dots & Q_{1,p-1} & r_1 \\ \vdots & \vdots & & & \vdots \\ Q_{p-1,0} & Q_{p-1,1} & \dots & Q_{p-1,p-1} & r_{p-1} \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix},$$

where \mathbf{Q} is the (p, p) matrix of transient probabilities, $\mathbf{0} = (0, 0, \dots, 0)^T$, and the p vector \mathbf{r} satisfies $\mathbf{r} = (1 - \mathbf{Q}\mathbf{1})$ (i.e., row probabilities must sum to 1) with $\mathbf{1} = (1, 1, \dots, 1)^T$. For the upward (downward) CUSUM control chart, the statistic C_i^+ (C_i^-) experiences p different transitional states before being absorbed into the out-of-control state. The states 0 to $(p - 1)$ are in-control states and state p is an out-of-control state. The width δ of the interval of each in-control state is given as $\delta = \frac{2H^+}{2p-1}$ ($\delta = \frac{2H^-}{2p-1}$). By definition, we have $H_j = j \times \delta, j = 0, 1, \dots, p - 1$. When the number p of subintervals is sufficiently large, this finite approach provides an effective method that allows the run-length properties (*ARL*) to be accurately evaluated. In this study, the generic element $Q_{i,j}, i = 0, 1, \dots, p - 1$ of the matrix \mathbf{Q} of transient probabilities is equal to the following:

- for the upward CUSUM control chart

$$\begin{aligned} Q_{i,0} &= F_{Y_k}(\mu_0 - H_i + K^+ + 0.5\delta), \quad (18.3) \\ Q_{i,j} &= F_{Y_k}(\mu_0 + H_j - H_i + 0.5\delta + K^+) \\ &\quad - F_{Y_k}(\mu_0 + H_j - H_i - 0.5\delta + K^+), \\ &\quad j = 1, \dots, p - 1; \end{aligned}$$

- for the downward CUSUM control chart

$$\begin{aligned} Q_{i,0} &= 1 - F_{Y_k}(\mu_0 + H_i - K^- - 0.5\delta), \quad (18.4) \\ Q_{i,j} &= F_{Y_k}(\mu_0 + H_i - H_j + 0.5\delta - K^-) \\ &\quad - F_{Y_k}(\mu_0 + H_i - H_j - 0.5\delta - K^-), \\ &\quad j = 1, \dots, p - 1, \end{aligned}$$

where $F_{Y_k}(\cdot)$ is the *c.d.f.* of Y_k . Let \mathbf{q} be the $(p - 1, 1)$ vector of initial probabilities associated with the p transient states, i.e., $\mathbf{q} = (q_0, q_1, \dots, q_{p-1})^T$. As proposed by Neuts [46] and Latouche [47], since the number of steps RL until the process reaches the absorbing state is a *discrete phase-type* random variable of parameters (\mathbf{Q}, \mathbf{q}) , the mean (*ARL*) and the standard-deviation (*SDRL*) of the RL of the one-sided CUSUM control charts are equal to

$$\begin{aligned} ARL &= \mathbf{q}^T (\mathbf{I} - \mathbf{Q})^{-1} \mathbf{1}, \quad (18.5) \\ SDRL &= \sqrt{2\mathbf{q}^T (\mathbf{I} - \mathbf{Q})^{-1} \mathbf{Q} \mathbf{1} - ARL^2 + ARL}. \quad (18.6) \end{aligned}$$

The ARL of the two-sided CUSUM control chart, denoted by $ARL_{two-sided}$, can be obtained from the ARL of the one-sided CUSUM charts by

$$\frac{1}{ARL_{two-sided}} = \frac{1}{ARL_U} + \frac{1}{ARL_D}, \quad (18.7)$$

where ARL_U and ARL_D stand for the ARL of the upward chart and the downward chart, respectively.

The Fredholm Integral Equation Method

Denote $L(y)$ the ARL for one-sided CUSUM scheme given that the initialization is $C_0^+ = y (C_0^- = y)$. According to Page [43] and Crowder [44], $L(y)$ can be defined in the following form:

$$L(y) = 1 + L(0)F_{Y_k}(-y) + \int_0^H L(s)f_{Y_k}(s - y)ds \quad (18.8)$$

By applying the Gauß-Legendre-Nyström approach, we can numerically solve the integral equation (18.8). In particular, the integration in (18.8) can be replaced by an appropriate quadrature, considering the equation only at the related nodes of the quadrature and solving the resulting linear equation system. Suppose that from the quadrature rule we obtain nodes s_1, s_2, \dots, s_N and weights w_1, w_2, \dots, w_N for given $N \in \mathbb{N}$.

The resulting linear equation system has the following form (for $i = 1, 2, \dots, N$):

$$\mathcal{L}(s_i) = 1 + L(0)F_{Y_k}(-y) + \sum_{j=1}^N w_j f_{Y_k}(s_j - s_i) \mathcal{L}(s_j). \quad (18.9)$$

Then, after solving this linear equation system, $\mathcal{L}(y)$ can be evaluated for arbitrary $y \in [0, H]$ by replacing s_i by y in (18.9). More details regarding this numerical procedure can be seen in a study by Crowder [44].

18.2 CUSUM Control Chart Monitoring the CV

As discussed by Castagliola et al. [8], monitoring the S^2 statistic using the CUSUM control chart is more efficient than monitoring directly the S statistic. Therefore, in this study, we will focus on monitoring the CV squared rather than monitoring the CV itself to obtain a control chart with high performance.

18.2.1 Basic Properties of Sample CV Squared

Let X be a positive random variable. The CV γ of the random variable X is defined as the ratio of its standard deviation $\sigma = \sigma(X)$ to its mean $\mu = E(X)$, that is

$$\gamma = \frac{\sigma}{\mu}.$$

Suppose that a set of *i.i.d.* samples $\{X_1, \dots, X_n\}$ is collected where each sample follows a normal distribution with parameters (μ, σ^2) . Let \bar{X} and S be the sample mean and the sample standard-deviation of X_1, \dots, X_n , i.e.

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i,$$

and

$$S = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2}.$$

Then the sample CV $\hat{\gamma}$ is defined as

$$\hat{\gamma} = \frac{S}{\bar{X}}.$$

In the literature, the distribution of the sample CV $\hat{\gamma}$ has been studied by many authors (see, e.g., McKay [48], Hendricks and Robey [49], Iglewicz and Myers [50], Iglewicz et al. [51], Vangel [52], Waren [53] and Reh and Scheffler [54]). However, the exact cumulative distribution function (*c.d.f.*) of $\hat{\gamma}$ has a complicated form. Furthermore, Iglewicz et al. [51] showed that if $0 < \gamma \leq 0.5$, $\sqrt{n}/\hat{\gamma}$ follows a noncentral t-distribution with $n - 1$ degrees of freedom and noncentrality parameter \sqrt{n}/γ . From this result, it is easy to prove that $n/\hat{\gamma}^2$ follows a noncentral F distribution with $(1, n - 1)$ degrees of freedom and noncentrality parameter n/γ^2 . Therefore, the *c.d.f.* of the sample CV squared $\hat{\gamma}^2$ can be derived as

- for $x \leq 0$, $F_{\hat{\gamma}^2}(x | n, \gamma) = 0$;
- for $x > 0$,

$$\begin{aligned} F_{\hat{\gamma}^2}(x | n, \gamma) &= P(\hat{\gamma}^2 \leq x | n, \gamma) \\ &= P\left(\frac{n}{\hat{\gamma}^2} \geq \frac{n}{x} > 0 | n, \gamma\right) \\ &= 1 - P\left(0 < \frac{n}{\hat{\gamma}^2} \leq \frac{n}{x} | n, \gamma\right) \\ &= 1 - F_F\left(\frac{n}{x} \mid 1, n - 1, \frac{n}{\gamma^2}\right), \end{aligned} \quad (18.10)$$

where $F_F(\cdot)$ is the *c.d.f.* of the noncentral F distribution. Inverting $F_{\hat{\gamma}^2}(x | n, \gamma)$ gives the inverse *c.d.f.* $F_{\hat{\gamma}^2}^{-1}(\alpha | n, \gamma)$ of $\hat{\gamma}^2$ as

$$F_{\hat{\gamma}^2}^{-1}(\alpha | n, \gamma) = \frac{n}{F_F^{-1}\left(1 - \alpha \mid 1, n - 1, \frac{n}{\gamma^2}\right)}, \quad (18.11)$$

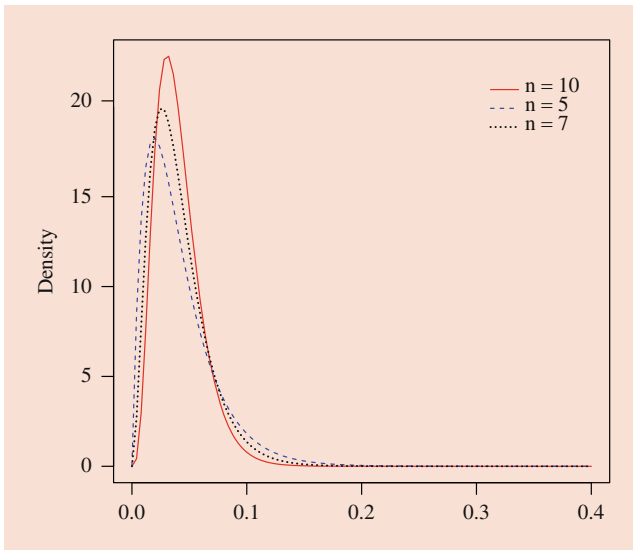


Fig. 18.2 The approximate density function of the sample CV squared for $\gamma = 0.2$

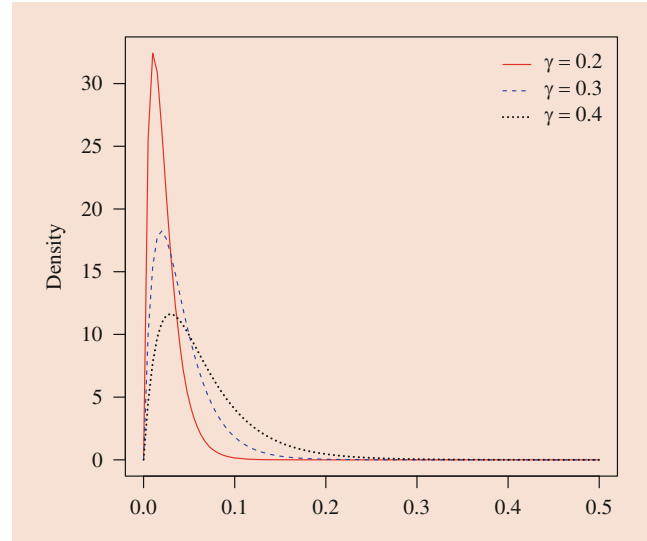


Fig. 18.3 The approximate density function of the sample CV squared for $n = 5$

where $F_F^{-1}(\cdot)$ is the inverse *c.d.f.* of the noncentral *F* distribution.

Figures 18.2 and 18.3 display the approximate density function of $\hat{\gamma}^2$ for several situations of parameters.

In the literature, there is no closed form for the mean and the standard deviation of $\hat{\gamma}^2$. Nevertheless, Breunig [55] provided an accurate approximation for $\mu_0(\hat{\gamma}^2)$ and $\sigma_0(\hat{\gamma}^2)$ as

$$\mu_0(\hat{\gamma}^2) = \gamma_0^2 \left(1 - \frac{3\gamma_0^2}{n} \right), \tag{18.12}$$

$$\sigma_0(\hat{\gamma}^2) = \sqrt{\gamma_0^4 \left(\frac{2}{n-1} + \gamma_0^2 \left(\frac{4}{n} + \frac{20}{n(n-1)} + \frac{75\gamma_0^2}{n^2} \right) \right) - (\mu_0(\hat{\gamma}^2) - \gamma_0^2)^2}. \tag{18.13}$$

18.2.2 Implementation and Design of CUSUM- γ^2 Control Charts

As can be seen from Figures 18.2 and 18.3, the distribution of the CV squared is skewed. Monitoring this variable using a two-sided control chart may lead to the *ARL*-biased property. That is to say, the out-of-control value ARL_1 can be larger than the in-control value ARL_0 . The main reason for this *ARL*-biased property is that in a two-sided control chart, the symmetrical control limits above and below the centerline are designed for the variables which follow an asymmetrical distribution. To avoid this situation, the one-sided control charts are proposed in the literature, that is, instead of monitoring both the increase and the decrease of the characteristic of interest simultaneously, each of these changes is monitored separately in different control charts. Castagliola et al. [8] also discussed several reasons for the advantage of using a one-sided control chart to monitor the CV, which is:

- It is able to control separately the in-control value ARL_0 from each one-sided chart.
- In general, the out-of-control tendency is usually known in advance and typically corresponds to an increase (or a decrease) in γ . Thus, it is possible to tune the upward and downward parts separately.

Therefore, we apply two one-sided CUSUM control charts to monitor the CV squared in this study. In particular, these two control charts (denoted by CUSUM- γ^2 from now in this chapter) involved:

- an upward CUSUM chart (denoted by “upward CUSUM- γ^2 ”) for detecting an increase in the CV

$$C_i^+ = \max\{0, C_{i-1}^+ + (\hat{\gamma}_i^2 - \mu_0(\hat{\gamma}^2)) - K^+\}, \tag{18.14}$$

with the initial values $C_0^+ = 0$, reference value $K^+ = k^+ \sigma_0(\hat{\gamma}^2)$, and corresponding upper control limit $H^+ = h^+ \mu_0(\hat{\gamma}^2)$

- a downward CUSUM chart (denoted by “downward CUSUM- γ^2 ”) for detecting a decrease in the CV

$$C_i^- = \max(0, C_{i-1}^- - (\hat{\gamma}_i^2 - \mu_0(\hat{\gamma}^2) - K^-), (18.15)$$

with the initial values $C_0^- = 0$, the reference value $K^- = k^- \sigma_0(\hat{\gamma}^2)$, and the corresponding upper control limit $H^- = h^- \mu_0(\hat{\gamma}^2)$.

The parameters (k^+, h^+) and (k^-, h^-) are the upward and downward CUSUM- γ^2 chart coefficients, respectively. These coefficients define a CUSUM- γ^2 control chart. The performance of the charts is evaluated by using the *ARL* measure which can be numerically calculated for a particular shift size τ . The values $\tau \in (0, 1)$ correspond to a decrease of the nominal CV, while the values $\tau > 1$ correspond to an increase of the nominal CV.

In general, the design of the CUSUM- γ^2 charts is implemented by finding out the optimal couples (k^+, h^+) or (k^-, h^-) that minimize the out-of-control *ARL*₁ for a given in-control value *ARL*₀ where the *ARL* value is calculated based on the formula (18.5) with the *c.d.f.* of $\hat{\gamma}^2$ is defined in (18.24). This procedure includes two main steps:

- Find the potential combinations (k^+, h^+) or (k^-, h^-) such that *ARL* = *ARL*₀, where *ARL*₀ is a predefined in-control *ARL* value.
- Among these combinations, choose an optimal one, say (k^{*+}, h^{*+}) or (k^{*-}, h^{*-}) , that gives the best performance, i.e., the smallest out-of-control *ARL* value for a particular shift τ , from an in-control value γ_0 to an out-of-control value $\gamma_1 = \tau \gamma_0$.

The above procedure is designed for a specific value of the shift size. However, it is usually not possible to predict exactly the shift size of τ in practice. A consequence of the wrong prediction where the actual process shift is different from the one used to design the control chart is that the designed control chart can have a poor performance. In order to overcome this problem, one can consider τ as a random variable and assume a statistical distribution to model it. If the quality practitioners prefer to get an optimal design of a control chart with respect to a range of shifts sizes $[a, b]$ without any preference for a specific size, a uniform distribution $f_\tau(\tau) = \frac{1}{b-a}$ can be used to give equal weight to each shift size included within the interval $[a, b]$. In this situation, the performance of control charts is evaluated by the expected average run length (*EARL*), which is defined by

$$EARL = \int_a^b ARL \times f_\tau(\tau) d\tau. \quad (18.16)$$

Therefore, the optimal coefficients (k^{*+}, h^{*+}) or (k^{*-}, h^{*-}) should satisfy the following conditions:

- for the downward CUSUM- γ^2 control chart

$$(k^{*-}, h^{*-}) = \operatorname{argmin}_{(k^-, h^-)} EARL(\gamma_0, \tau, k^-, h^-, n) \quad (18.17)$$

subject to the constraint

$$EARL(\gamma_0, \tau = 1, k^-, h^-, n) = ARL(\gamma_0, \tau = 1, k^-, h^-, n) = ARL_0, \quad (18.18)$$

- for the upward CUSUM- γ^2 control chart

$$(k^{*+}, h^{*+}) = \operatorname{argmin}_{(k^+, h^+)} EARL(\gamma_0, \tau, k^+, h^+, n) \quad (18.19)$$

subject to the constraint

$$EARL(\gamma_0, \tau = 1, k^+, h^+, n) = ARL(\gamma_0, \tau = 1, k^+, h^+, n) = ARL_0. \quad (18.20)$$

18.2.3 Performance of the CUSUM- γ^2 Control Charts

The performance of the CUSUM- γ^2 control charts and a direct comparison with other control charts monitoring the CV squared have been investigated by Tran and Tran [21]. Given the value of n , γ_0 , the *ARL* and *SDRL* of the CUSUM- γ^2 control charts can be calculated with respect to a specific shift size following the formulas (18.5)–(18.6). The optimal parameters (k^{*+}, h^{*+}) and (k^{*-}, h^{*-}) and the corresponding (*ARL*, *SDRL*) values are presented in Tables 18.2 and 18.3 for the shift size $\tau = \{0.5, 0.65, 0.8, 1.25, 1.5, 2.0\}$, in-control CV $\gamma_0 = \{0.05, 0.1, 0.15, 0.2\}$, sample size $n = \{5, 7\}$ (Table 18.2) and $n = \{10, 15\}$ (Table 18.3). In the optimization procedure, the in-control *ARL* is set at *ARL*₀ = 370.4.

It can be seen from the obtained results that given τ and n , the value of γ_0 has a trivial impact on the performance of the one-sided CUSUM- γ^2 control charts. When γ_0 increases from 0.05 to 0.2, the values of both *ARL*₁ and *SDRL*₁ change insignificantly. For example, with $n = 5$, $\tau = 0.8$ we have *ARL*₁ = 20.0, *SDRL*₁ = 10.0 when $\gamma_0 = 0.05$ and *ARL*₁ = 20.4, *SDRL*₁ = 10.4 when $\gamma_0 = 0.2$. By contrast, the performance of the CUSUM- γ^2 control charts depends strongly on the sample size n : the larger the value of n , the smaller the value of the *ARL*₁. For example, when $\tau = 1.25$, $\gamma_0 = 0.1$, we have *ARL*₁ = 14.8 if $n = 5$, and we have *ARL*₁ = 6.0 if $n = 15$.

Table 18.4 shows the overall performance of the CUSUM- γ^2 control charts according to the *EARL* measure. The unique optimal couples (k^{*+}, h^{*+}) and (k^{*-}, h^{*-}) are found satisfying equations (18.17) and (18.19) subject to the constraints (18.18) and (18.20) where the support of τ is $\Omega_\tau = (1, 2]$

Table 18.2 The optimal couples (k^{*+}, h^{*+}) when $\tau = \{0.5, 0.65, 0.8, 0.9\}$, (k^{*-}, h^{*-}) when $\tau = \{1.1, 1.25, 1.5, 2\}$ (first row) and the corresponding $(ARL_1, SDR L_1)$ values (second row) of the CUSUM- γ^2 charts for $n = \{5, 7\}$ and $\gamma_0 = \{0.05, 0.1, 0.15, 0.2\}$

τ	$\gamma_0 = 0.05$	$\gamma_0 = 0.1$	$\gamma_0 = 0.15$	$\gamma_0 = 0.2$
$n = 5$				
0.50	(0.7474, 0.8105) (4.3, 1.5)	(0.7383, 0.7915) (4.3, 1.5)	(0.7124, 0.7980) (4.3, 1.5)	(0.6762, 0.8127) (4.3, 1.6)
0.65	(0.5203, 1.6283) (8.2, 3.5)	(0.5044, 1.6444) (8.2, 3.5)	(0.4805, 1.6731) (8.3, 3.5)	(0.4493, 1.7126) (8.3, 3.5)
0.80	(0.2864, 3.3713) (20.0, 10.0)	(0.2740, 3.3938) (20.0, 10.1)	(0.2532, 3.4601) (20.2, 10.2)	(0.2256, 3.5586) (20.4, 10.4)
1.25	(0.0683, 4.8431) (14.7, 10.1)	(0.0556, 4.9781) (14.8, 10.2)	(0.0352, 5.2355) (15.0, 10.4)	(0.0081, 5.6251) (15.4, 10.6)
1.50	(0.0683, 3.5041) (5.6, 3.8)	(0.0556, 3.6291) (5.6, 3.8)	(0.0352, 3.8568) (5.7, 3.9)	(0.0081, 4.1954) (5.9, 4.0)
2.00	(0.0683, 2.5174) (2.3, 1.5)	(0.0556, 2.6300) (2.3, 1.5)	(0.0352, 2.8313) (2.4, 1.6)	(0.0081, 3.1351) (2.5, 1.6)
$n = 7$				
0.50	(0.9283, 0.5307) (3.1, 1.1)	(0.9105, 0.5274) (3.1, 1.1)	(0.8844, 0.5300) (3.1, 1.1)	(0.8459, 0.5421) (3.1, 1.1)
0.65	(0.6349, 1.1324) (5.9, 2.4)	(0.6182, 1.1400) (5.9, 2.4)	(0.5989, 1.1446) (6.0, 2.5)	(0.5701, 1.1631) (6.0, 2.5)
0.80	(0.3582, 2.3581) (14.8, 7.4)	(0.3438, 2.3851) (14.8, 7.4)	(0.3235, 2.4313) (15.0, 7.5)	(0.2961, 2.5021) (15.2, 7.6)
1.25	(0.0870, 3.4343) (11.0, 7.3)	(0.0744, 3.5308) (11.1, 7.4)	(0.0555, 3.6938) (11.3, 7.5)	(0.0296, 3.9375) (11.6, 7.7)
1.50	(0.0870, 2.4048) (4.1, 2.7)	(0.0744, 2.4893) (4.2, 2.7)	(0.0555, 2.6311) (4.3, 2.8)	(0.0296, 2.8450) (4.4, 2.9)
2.00	(0.0870, 1.6354) (1.8, 1.0)	(0.0744, 1.7075) (1.8, 1.0)	(0.0555, 1.8323) (1.8, 1.1)	(0.0296, 2.0195) (1.9, 1.1)

(increasing case) and $\Omega_D = [0.5, 1)$ (decreasing case), respectively. The values of γ_0 and n are the same as the ones in Tables 18.2 and 18.3. A similar tendency of the dependence of the CUSUM- γ^2 control charts' performance on γ_0 and n can be seen in Table 18.4: the $EARL$ does not change much when γ_0 varies, but it decreases significantly when n increases.

Example

Table 18.5 presents a real phase II dataset from a sintering process in an Italian company that manufactures sintered mechanical parts, which is introduced by Castagliola et al. [8]. Throughout the sintering process, molten copper is used to filling pores, allowing the drop time to be prolonged notably. Moreover, the larger the quantity Q_C of molten copper absorbed within the sintered compact during cooling is, the more the expected pressure drop time T_{pd} is. It is stated that the preliminary regression study relating T_{pd} to the quantity Q_C of molten copper has demonstrated the presence of a constant proportionality $\sigma_{pd} = \gamma_{pd} \times \mu_{pd}$ between the standard deviation of the pressure drop time and its mean. The CV $\gamma_{pd} = \sigma_{pd}/\mu_{pd}$ is then monitored to detect changes in the process variability. The recorded data consists of 20

new samples taken from the process after the occurrence of a special cause increasing process variability.

The upward CUSUM- γ^2 control chart is designed for the unknown shift size case. From the Phase I data, the in-control CV has been estimated as $\hat{\gamma}_0 = 0.417$. The in-control mean and variance are calculated from (18.12) and (18.13) as $\mu_0 = 0.15575$ and $\sigma_0 = 0.164307$. The chart coefficients which are optimal for detecting a shift with a uniform distribution over the interval $(1, 2)$ are found to be $k^{*+} = 0.3898930$ and $h^{*+} = 12.264137$. Then, we obtain $K^+ = 0.064062$ and $H^+ = 1.910097$. The values of C_i^* are then presented in the rightmost column in Table 18.5 and plotted in Fig. 18.4 along with the control limit H^+ . As can be seen from this figure, the CUSUM- γ^2 chart detects several out-of-control samples (in bold in Table 18.5), from the 13th sample onward.

18.3 CUSUM Control Chart Monitoring the CV in the Presence of Measurement Error

The CUSUM- γ^2 control charts designed above are based on an assumption that the quality of interest X_i can be measured

Table 18.3 The optimal couples (k^{*+}, h^{*+}) when $\tau = \{0.5, 0.65, 0.8, 0.9\}$, (k^{*-}, h^{*-}) when $\tau = \{1.1, 1.25, 1.5, 2\}$ (first row) and the corresponding $(ARL_1, SDR L_1)$ values (second row) of the CUSUM- γ^2 charts for $n = \{10, 15\}$ and $\gamma_0 = \{0.05, 0.1, 0.15, 0.2\}$

τ	$\gamma_0 = 0.05$	$\gamma_0 = 0.1$	$\gamma_0 = 0.15$	$\gamma_0 = 0.2$
$n = 10$				
0.50	(0.9385, 0.5557) (2.4, 0.6)	(1.1201, 0.3581) (2.3, 0.8)	(1.0939, 0.3526) (2.3, 0.8)	(1.0558, 0.3554) (2.3, 0.8)
0.65	(0.7737, 0.7932) (4.4, 1.8)	(0.7552, 0.7884) (4.3, 1.7)	(0.7440, 0.7761) (4.3, 1.8)	(0.7129, 0.7917) (4.4, 1.8)
0.80	(0.4318, 1.7024) (10.9, 5.3)	(0.4271, 1.6708) (10.9, 5.3)	(0.4036, 1.7169) (11.0, 5.3)	(0.3824, 1.7415) (11.2, 5.5)
1.25	(0.1094, 2.4092) (8.2, 5.3)	(0.0966, 2.4735) (8.2, 5.3)	(0.0799, 2.5811) (8.4, 5.4)	(0.0580, 2.7378) (8.6, 5.6)
1.50	(0.9729, 1.6244) (3.1, 1.9)	(0.0966, 1.6786) (3.1, 1.9)	(0.0799, 1.7719) (3.2, 2.0)	(0.0580, 1.9056) (3.3, 2.1)
2.00	(1.7924, 1.0143) (1.4, 0.6)	(1.7732, 1.0605) (1.4, 0.7)	(0.0799, 1.1389) (1.5, 0.7)	(0.0580, 1.2541) (1.5, 0.8)
$n = 15$				
0.50	(0.8349, 0.6833) (2.1, 0.3)	(1.2498, 0.3121) (1.7, 0.6)	(1.2087, 0.3111) (1.6, 0.6)	(1.1998, 0.2973) (1.6, 0.6)
0.65	(0.8349, 0.6833) (3.2, 1.1)	(0.7872, 0.6981) (3.1, 1.0)	(0.9395, 0.5133) (3.1, 1.2)	(0.9109, 0.5143) (3.1, 1.2)
0.80	(0.5328, 1.1938) (7.9, 3.7)	(0.5183, 1.1810) (7.8, 3.6)	(0.5124, 1.1619) (7.9, 3.7)	(0.4890, 1.1855) (8.0, 3.8)
1.25	(0.1387, 1.6150) (5.9, 3.7)	(0.6337, 1.6566) (6.0, 3.7)	(0.1115, 1.7265) (6.1, 3.8)	(0.0917, 1.8239) (6.2, 3.9)
1.50	(0.1387, 1.0351) (2.2, 1.3)	(1.2037, 1.0688) (2.2, 1.3)	(0.1115, 1.5118) (2.4, 1.2)	(1.1702, 1.2059) (2.3, 1.4)
2.00	(0.1387, 1.4481) (1.2, 0.4)	(2.2136, 0.5695) (1.1, 0.4)	(2.1751, 0.6188) (1.1, 0.4)	(2.1238, 0.6897) (1.2, 0.5)

Table 18.4 The control coefficients (k^{*+}, h^{*+}) for $\Omega_D = [0.5, 1)$ and (k^{*-}, h^{*-}) for $\Omega_I = (1, 2]$ (first row) and the corresponding EARL values (second row) of the CUSUM- γ^2 control charts for $n = \{5, 7, 10, 15\}$, $\gamma_0 = \{0.05, 0.1, 0.15, 0.2\}$

Ω	$\gamma_0 = 0.05$	$\gamma_0 = 0.1$	$\gamma_0 = 0.15$	$\gamma_0 = 0.2$
$n = 5$				
D	(0.1880, 4.8034) 28.9	(0.1743, 4.8684) 29.0	(0.1544, 4.9673) (29.2)	(0.1270, 5.1334) 29.5
I	(0.2107, 6.0442) 15.6	(0.2194, 6.2329) 15.7	(0.2366, 6.5034) 15.8	(0.2576, 6.9412) 16.1
$n = 7$				
D	(0.2124, 3.7306) 22.9	(0.1995, 3.7721) 22.9	(0.1808, 3.8468) 23.1	(0.1568, 3.9463) 23.4
I	(0.2321, 4.6091) 12.4	(0.2422, 4.7089) 12.5	(0.2559, 4.8963) 12.6	(0.2734, 5.1788) 12.9
$n = 10$				
D	(0.2278, 3.0091) 18.0	(0.2264, 2.9250) 18.0	(0.2114, 2.9581) 18.2	(0.1898, 3.0278) 18.4
I	(0.2608, 3.4709) 9.7	(0.2660, 3.5625) 9.8	(0.2805, 3.6630) 10.0	(0.2939, 3.8532) 10.2
$n = 15$				
D	(0.2630, 2.2677) 13.8	(0.2595, 2.2123) 13.7	(0.2480, 2.2191) 13.8	(0.2300, 2.2591) 14.0)
I	(0.2951, 2.5435) 7.4	(0.3035, 2.5828) 7.5	(0.3112, 2.6676) 7.6	(0.3227, 2.7825) 7.8

Table 18.5 Phase II datasets from a sintering process introduced by Castagliola et al. [8]

Phase II	\bar{X}_i	S_i	$\hat{\gamma}$	$\hat{\gamma}_i^2$	$1C_i^+$
1	906.4	476.0	0.525	0.27563	0.055816
2	805.1	493.9	0.614	0.37700	0.213004
3	1187.2	1105.9	0.932	0.86862	0.861819
4	663.4	304.8	0.459	0.21068	0.852691
5	1012.1	367.4	0.363	0.13177	0.764652
6	863.2	350.4	0.406	0.16484	0.709679
7	1561.0	1562.2	1.058	1.11936	1.609234
8	697.1	253.2	0.363	0.13177	1.521195
9	1024.6	120.9	0.118	0.01392	1.315310
10	355.3	235.2	0.662	0.43824	1.533745
11	485.6	106.5	0.219	0.04796	1.361898
12	1224.3	915.4	0.748	0.55950	1.701593
13	1365.0	1051.6	0.770	0.59290	2.074684
14	704.0	449.7	0.639	0.40832	2.263197
15	1584.7	1050.8	0.663	0.43957	2.482957
16	1130.0	680.6	0.602	0.36240	2.625552
17	824.7	393.5	0.477	0.22753	2.633273
18	921.2	391.6	0.425	0.18062	2.594089
19	870.3	730.0	0.839	0.70392	3.078201
20	1068.3	150.8	0.141	0.01988	2.878274

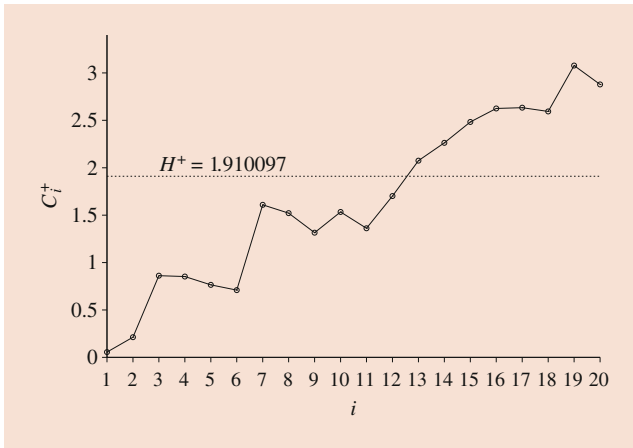


Fig. 18.4 CUSUM- γ^2 charts applied to the sintering process (Phase II)

exactly, i.e., without any measurement error. However, this assumption may not be reached in practice since the ME is almost ineluctable no matter how exact the measurement system is. It is proven within the industrial context that the ME may affect the performance of SPC methodologies. Since Bennet [56] investigated the effect of ME on the Shewhart \bar{X} chart, the topic has been widely studied in the SPC literature. Linna and Woodall [57] suggested a linear covariate error model to represent the relation between the observed value and the true value. Following this model, Maravelakis [58] and Maravelakis et al. [59] presented the effect of ME on the EWMA and CUSUM \bar{X} control chart, respectively.

Noorossana and Zerehsaz [60] discussed a problem of ME on the control chart monitoring the linear profiles. Tran et al. [61] provided a synthetic median control chart for monitoring the process mean with ME. The performance of a number of CV control charts in the presence of ME has also been recently studied by Nguyen et al. [37] and Tran et al. [36, 38, 62].

18.3.1 Linear Covariate Error Model for the Coefficient of Variation

In this section, we present briefly the linear covariate error model for the CV as suggested by Tran et al. [62]. Let $\{X_{i,1}, X_{i,2}, \dots, X_{i,n}\}$ be a set of samples of the quality of interest, $i = 1, 2, \dots$. The variable X_{ij} refers to the j^{th} sample taken at times i . Suppose that X_{ij} follows a normal distribution with mean $\mu_0 + a\sigma_0$ and standard deviation $b\sigma_0$. The constants a and b represent a shift of the process. If $a = 0$ and $b = 1$, the process is said to be in-control; otherwise, it is considered to be out-of-control.

Because of the ME problem, the true value of X_{ij} is not observable. Instead, this value can only be assessed through m real observations $\{X_{ij,1}^*, X_{ij,2}^*, \dots, X_{ij,m}^*\}$, $m \geq 1$. Linna and Woodall [57] suggested the following linearly covariate error model:

$$X_{ij,k}^* = A + BX_{ij} + \varepsilon_{ij,k},$$

where $X_{ij,k}^*$ is the observed value in the k^{th} measurement of the item j at the sampling i , A and B are two constants depending on the gauge location error, and $\varepsilon_{ij,k} \sim N(0, \sigma_M)$ is the random error term due to the gauge precision error which is independent of X_{ij} . In particular, A is the constant bias component, B is the parameter modeling the linearity error, and σ_M is the precision error. Figure 18.5 presents a graphical representation of the ME model.

Then, the mean

$$\bar{X}_{ij}^* = \frac{1}{m} \sum_{k=1}^m X_{ij,k}^* = A + BX_{ij} + \frac{1}{m} \sum_{k=1}^m \varepsilon_{ij,k}$$

of m observed quantities of the same item j is considered as a representation of X_{ij} . By its definition, \bar{X}_{ij}^* also follows a normal distribution with parameters

$$\begin{cases} \mu^* = A + B(\mu_0 + a\sigma_0), \\ \sigma^{*2} = B^2b^2\sigma_0^2 + \frac{\sigma_M^2}{m}. \end{cases}$$

Thus, the CV of the measured quantity \bar{X}_{ij}^* is defined by

$$\gamma^* = \frac{\sqrt{B^2b^2\sigma_0^2 + \frac{\sigma_M^2}{m}}}{A + B(\mu_0 + a\sigma_0)} = \frac{\sqrt{B^2b^2 + \frac{\eta^2}{m}}}{\theta + B(1 + a\gamma_0)} \times \gamma_0, \quad (18.21)$$

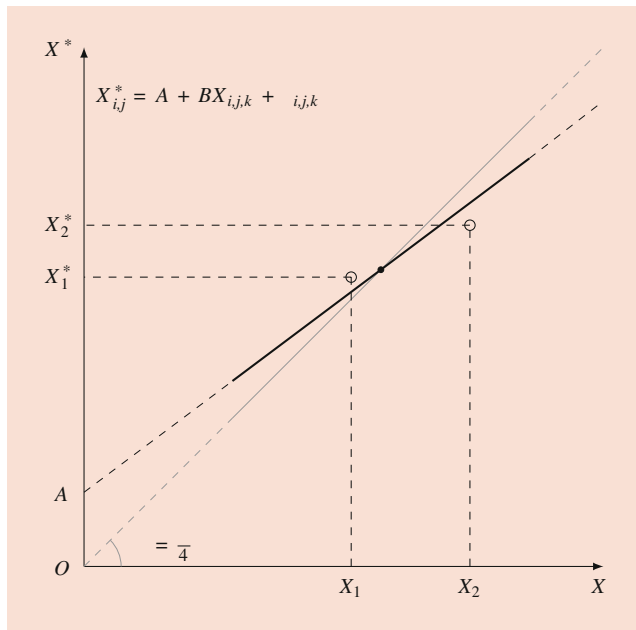


Fig. 18.5 A representation of the measurement error model

where $\eta = \frac{\sigma_M}{\sigma_0}$, $\gamma_0 = \frac{\sigma_0}{\mu_0}$, $\theta = \frac{A}{\mu_0}$; η and θ represent the precision error ratio and the accuracy error ratio.

It is important to consider that the ME also affects the relation between the in-control and the out-of-control values of the CV. Indeed, without ME, we have $\gamma_1 = \tau \gamma_0$ where τ is the true shift size. With the introduction of a and b , the out-of-control CV is defined by

$$\gamma_1 = \frac{b\sigma_0}{\mu_0 + a\sigma_0} = \frac{b}{1 + a\gamma_0} \gamma_0.$$

Then, the shift size τ can be expressed by $\tau = \frac{b}{1 + a\gamma_0}$. Under the presence of ME, the in-control value of the CV (corresponding to $a = 0$ and $b = 1$) now becomes

$$\gamma_0^* = \frac{\sqrt{B^2 + \frac{\eta^2}{m}}}{\theta + B} \times \gamma_0, \tag{18.22}$$

and the out-of-control value is

$$\gamma_1^* = \frac{\sqrt{B^2 b^2 + \frac{\eta^2}{m}}}{\theta + Bb/\tau} \times \gamma_0. \tag{18.23}$$

Equations (18.22)–(18.23) show that in general $\gamma_1^* \neq \tau \gamma_0^*$.

Let \bar{X}_i^* and S_i^* denote the sample mean and the sample standard deviation of $\bar{X}_{1j}^*, \dots, \bar{X}_{nj}^*$, i.e.,

$$\bar{X}_i^* = \frac{1}{n} \sum_{j=1}^n \bar{X}_{ij}^* \text{ and } S_i^* = \sqrt{\frac{1}{n-1} \sum_{j=1}^n (\bar{X}_{ij}^* - \bar{X}_i^*)^2}.$$

Then, the sample CV squared in the presence of ME, $\hat{\gamma}_i^*$, is defined by

$$\hat{\gamma}_i^* = \frac{S_i^*}{\bar{X}_i^*}.$$

From (18.24) and (18.25), the *c.d.f.* and *i.d.f.* of $\hat{\gamma}^{*2}$ are defined as

$$F_{\hat{\gamma}^{*2}}(x | n, \gamma) = 1 - F_F\left(\frac{n}{x} \mid 1, n-1, \frac{n}{\gamma^{*2}}\right), \tag{18.24}$$

where $F_F(\cdot)$ is the *c.d.f.* of the noncentral F distribution, and

$$F_{\hat{\gamma}^{*2}}^{-1}(\alpha | n, \gamma) \simeq \frac{n}{F_F^{-1}\left(1 - \alpha \mid 1, n-1, \frac{n}{\gamma^{*2}}\right)}, \tag{18.25}$$

where $F_F^{-1}(\cdot)$ is the inverse *c.d.f.* of the noncentral F distribution.

18.3.2 The Effect of Measurement Error on the CUSUM- γ^2 Control Charts

In a recent study on the effect of measurement error on the performance of the CUSUM- γ^2 control chart (Tran et al. [38]), the authors suggested using globally optimized coefficients which optimize the chart's performance in an anticipated interval of the shift size τ . In this study, we consider designing this control chart under the presence of ME for both cases of specific shift sizes and unknown shift sizes.

With the presence of ME, the implementation of the CUSUM- γ^2 control charts is similar to the one in Sect. 18.2.2. The optimal control chart coefficients (k^{*+}, h^{*+}) and (k^{*-}, h^{*-}) are also the solutions of the optimal problem of the function ARL (for a specific shift size) or $EARL$ (for the unknown shift size which is considered within a predetermined range) subject to a constraint on the in-control value ARL_0 . The difference is that the value ARL or $EARL$ should be calculated by using the distribution of γ^{*2} rather than the distribution of γ^2 .

To investigate the effect of ME on the CUSUM- γ^2 control charts' performance, we set the in-control value ARL_0 at 370.4. Without loss of generality, the shift in the variance is assumed to be a unit, i.e., $b = 1$. Several scenarios of other parameters are considered, where $\gamma_0 = \{0.05, 0.1, 0.2\}$, $n = \{5, 7, 10, 15\}$, $m = \{1, 3, 5, 7, 10\}$. Moreover, following the guidelines in the AIAG manual [63] for measurement system analysis, the value of η and θ is chosen less than 0.3, and the value of B is considered within the range [0.8, 1.2].

Given the values of m, n, B, η, θ , and γ_0 , the optimal couples (k^{*-}, h^{*-}) for the downward chart and (k^{*+}, h^{*+}) for the upward chart are solved. Tables 18.6 and 18.7 show the reference coefficients k^{*+}, k^{*-} (first row for each value of τ) and the limit coefficients h^{*+}, h^{*-} (second row for

Table 18.6 The effect of η on the chart coefficients k^* (first row) and h^* (second row) for $\gamma_0 = 0.05$ (left side), $\gamma_0 = 0.1$ (middle), $\gamma_0 = 0.2$ (right side), $\theta = 0.05$, $B = 1$, $m = 1$ and $n \in \{5, 7\}$

τ	$\gamma_0 = 0.05$												$\gamma_0 = 0.1$						$\gamma_0 = 0.2$																																																																																																																																																																																																																																																																																																																																																																																																										
	$\eta = 0$		$\eta = 0.05$		$\eta = 0.1$		$\eta = 0.15$		$\eta = 0.2$		$\eta = 0.25$		$\eta = 0$		$\eta = 0.05$		$\eta = 0.1$		$\eta = 0.15$		$\eta = 0.2$		$\eta = 0.25$																																																																																																																																																																																																																																																																																																																																																																																																						
	$n = 5$	0.8659	0.8655	0.8660	0.8381	0.8391	0.8412	0.8444	0.8449	0.8450	0.8458	0.8488	0.8634	0.8634	0.8633	0.8653	0.8717	0.8990	0.7272	0.7275	0.7270	0.7370	0.7355	0.7308	0.7182	0.7182	0.7282	0.7201	0.7193	0.7184	0.7153	0.7016	0.6648	0.6644	0.6588	0.6471	0.5958	1.7122	1.7116	1.7130	1.7121	1.7133	1.7182	1.7182	1.7289	1.7289	1.7310	1.7339	1.7507	1.8076	1.8078	1.8090	1.8104	1.8178	1.8922	3.4596	3.4617	3.4613	3.4623	3.4651	3.4754	3.5082	3.5049	3.5052	3.5093	3.5202	3.5612	3.6788	3.6810	3.6874	3.6990	3.7386	3.9399	0.2793	0.2791	0.2790	0.2787	0.2778	0.2743	0.2655	0.2657	0.2653	0.2644	0.2616	0.2500	0.2201	0.2195	0.2178	0.2150	0.2059	0.1645	4.9820	4.9821	4.9643	4.9688	4.9751	5.0077	5.0986	5.0997	5.1046	5.1138	5.1425	5.2854	5.6819	5.6906	5.7160	5.7573	5.8964	6.6070	0.3188	0.3189	0.3215	0.3213	0.3220	0.3242	0.3291	0.3293	0.3297	0.3302	0.3318	0.3387	0.3573	0.3576	0.3586	0.3605	0.3660	0.3910	3.6174	3.6185	3.6178	3.6218	3.6267	3.6568	3.7358	3.7387	3.7414	3.7499	3.7763	3.8991	4.2494	4.2597	4.2780	4.3176	4.4382	5.0621	0.6080	0.6078	0.6085	0.6078	0.6085	0.6089	0.6104	0.6100	0.6108	0.6110	0.6132	0.6132	0.6185	0.6178	0.6192	0.6191	0.6210	0.6291	2.6346	2.6361	2.6372	2.6374	2.6450	2.6717	2.7398	2.7412	2.7443	2.7524	2.7753	2.8852	3.1971	3.2043	3.2235	3.2568	3.3671	3.9277	1.1033	1.1025	1.1024	1.1033	1.1014	1.0989	1.0951	1.1025	1.0955	1.0943	1.0926	1.0846	1.0653	1.0645	1.0638	1.0619	1.0551	1.0296	$n = 7$	0.5765	0.5769	0.5772	0.5651	0.5782	0.5695	0.5729	0.5612	0.5608	0.5615	0.5630	0.5767	0.5770	0.5770	0.5771	0.5777	0.5927	0.9001	0.8994	0.8989	0.9062	0.8966	0.8961	0.8832	0.8903	0.8897	0.8895	0.8858	0.8318	0.8309	0.8291	0.8259	0.8158	0.7647	1.2333	1.2336	1.1984	1.1987	1.1994	1.2026	1.1828	1.2064	1.2065	1.2064	1.2073	1.2100	1.2226	1.2233	1.2245	1.2283	1.2536	1.2774	0.6022	0.6019	0.6131	0.6127	0.6116	0.6065	0.6054	0.5975	0.5971	0.5966	0.5941	0.5837	0.5565	0.5558	0.5540	0.5506	0.5360	0.4980	2.4548	2.4566	2.4562	2.4571	2.4599	2.4653	2.4877	2.4850	2.4847	2.4753	2.4908	2.5194	2.5947	2.5953	2.5980	2.6044	2.6289	2.7497	0.3449	0.3446	0.3445	0.3441	0.3431	0.3394	0.3305	0.3307	0.3304	0.3313	0.3274	0.3161	0.2878	0.2874	0.2860	0.2834	0.2748	0.2359	3.5393	3.5395	3.5400	3.5403	3.5440	3.5655	3.6244	3.6260	3.6297	3.6353	3.7443	3.9993	3.9998	3.9998	4.0176	4.0435	4.1315	4.5605	0.3916	0.3917	0.3918	0.3922	0.3928	0.3945	0.3981	0.3981	0.3983	0.3987	0.4000	0.4177	0.4188	0.4192	0.4206	0.4242	0.4423	0.4423	2.4947	2.4949	2.4963	2.4971	2.5007	2.5192	2.5707	2.5729	2.5763	2.5805	2.5974	2.6764	2.8981	2.9011	2.9152	2.9386	3.0143	3.3897	0.7438	0.7438	0.7435	0.7437	0.7440	0.7442	0.7435	0.7429	0.7427	0.7431	0.7427	0.7418	0.7391	0.7398	0.7395	0.7392	0.7382	0.7363	1.7308	1.7314	1.7311	1.7327	1.7372	1.7518	1.7962	1.7980	1.8005	1.8048	1.8206	1.8886	2.0841	2.0870	2.0988	2.1198	2.1866	2.5207	1.3504	1.3499	1.3509	1.3501	1.3484	1.3477	1.3401	1.3390	1.3388	1.3381	1.3343	1.3238	1.2927	1.2933	1.2918	1.2884	1.2789

Table 18.8 The effect of the precision error ratio η on the CUSUM- γ^2 charts' performance for $\gamma_0 = 0.05$ (left side), $\gamma_0 = 0.1$ (middle), $\gamma_0 = 0.2$ (right side), $\theta = 0.05$, $B = 1$ and $m = 1$

τ	$\gamma_0 = 0.05$						$\gamma_0 = 0.1$						$\gamma_0 = 0.2$					
	$\eta = 0$	$\eta = 0.05$	$\eta = 0.1$	$\eta = 0.15$	$\eta = 0.2$	$\eta = 0.25$	$\eta = 0$	$\eta = 0.05$	$\eta = 0.1$	$\eta = 0.15$	$\eta = 0.2$	$\eta = 0.25$	$\eta = 0$	$\eta = 0.05$	$\eta = 0.1$	$\eta = 0.15$	$\eta = 0.2$	$\eta = 0.25$
$n = 5$																		
0.5	4.51	4.52	4.52	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.55	4.55	4.55	4.55	4.55
0.7	8.66	8.66	8.66	8.66	8.66	8.66	8.67	8.67	8.67	8.67	8.67	8.67	8.79	8.79	8.79	8.79	8.79	8.80
0.8	21.1	21.1	21.1	21.1	21.1	21.1	21.2	21.2	21.2	21.2	21.2	21.2	21.5	21.5	21.5	21.5	21.5	21.5
1.3	15.9	15.9	15.9	15.9	15.9	15.9	16.0	16.0	16.0	16.0	16.0	16.1	16.6	16.6	16.6	16.6	16.6	16.6
1.5	6.13	6.13	6.13	6.13	6.13	6.13	6.19	6.19	6.19	6.19	6.20	6.20	6.46	6.46	6.46	6.47	6.47	6.48
2.0	2.57	2.57	2.57	2.57	2.57	2.57	2.60	2.60	2.60	2.60	2.61	2.61	2.74	2.74	2.75	2.75	2.75	2.76
$n = 7$																		
0.5	3.26	3.26	3.26	3.26	3.26	3.26	3.23	3.23	3.23	3.23	3.23	3.23	3.26	3.25	3.26	3.25	3.26	3.26
0.7	6.29	6.29	6.28	6.29	6.28	6.28	6.27	6.27	6.27	6.27	6.27	6.27	6.37	6.37	6.37	6.37	6.37	6.37
0.8	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	16.0	16.0	16.0	16.0	16.0	16.0
1.3	11.9	11.9	11.9	11.9	11.9	11.9	12.0	12.0	12.0	12.0	12.0	12.0	12.5	12.5	12.5	12.5	12.5	12.5
1.5	4.54	4.54	4.54	4.54	4.54	4.54	4.59	4.59	4.59	4.60	4.60	4.60	4.82	4.82	4.82	4.83	4.83	4.84
2.0	1.95	1.95	1.95	1.95	1.95	1.95	1.98	1.98	1.98	1.98	1.98	1.98	2.09	2.09	2.10	2.10	2.10	2.10
$n = 10$																		
0.5	2.47	2.47	2.47	2.47	2.47	2.47	2.44	2.44	2.44	2.44	2.44	2.44	2.40	2.40	2.41	2.40	2.40	2.40
0.7	4.62	4.61	4.61	4.61	4.61	4.61	4.55	4.55	4.55	4.55	4.55	4.55	4.60	4.60	4.61	4.61	4.61	4.61
0.8	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.8	11.8	11.8	11.8	11.8	11.87
1.3	8.90	8.90	8.90	8.90	8.90	8.90	8.99	8.99	8.99	8.99	9.00	9.00	9.38	9.38	9.38	9.39	9.40	9.41
1.5	3.36	3.36	3.36	3.36	3.36	3.36	3.40	3.40	3.41	3.41	3.41	3.41	3.59	3.59	3.59	3.59	3.60	3.60
2.0	1.52	1.52	1.52	1.52	1.52	1.52	1.54	1.54	1.54	1.54	1.54	1.54	1.62	1.62	1.62	1.62	1.63	1.63
$n = 15$																		
0.5	2.14	2.13	2.13	2.13	2.13	2.13	1.77	1.73	1.82	1.74	1.74	1.75	1.68	1.78	1.67	1.68	1.76	1.75
0.7	3.40	3.40	3.40	3.39	3.40	3.40	3.30	3.30	3.30	3.30	3.30	3.29	3.29	3.29	3.29	3.29	3.30	3.30
0.8	8.48	8.47	8.47	8.46	8.47	8.47	8.33	8.33	8.34	8.33	8.34	8.33	8.46	8.46	8.46	8.47	8.47	8.48
1.3	6.41	6.41	6.41	6.41	6.41	6.41	6.47	6.47	6.47	6.47	6.48	6.48	6.77	6.78	6.78	6.78	6.79	6.80
1.5	2.43	2.43	2.43	2.43	2.43	2.43	2.46	2.46	2.46	2.47	2.47	2.47	2.60	2.60	2.60	2.61	2.61	2.61
2.0	1.21	1.21	1.21	1.21	1.21	1.21	1.22	1.22	1.22	1.22	1.22	1.23	1.28	1.28	1.28	1.28	1.28	1.28

Table 18.9 The effect of the accuracy error θ on the CUSUM- γ^2 charts' performance for $\gamma_0 = 0.05$ (left side), $\gamma_0 = 0.1$ (middle), $\gamma_0 = 0.2$ (right side), $\eta = 0.28$, $B = 1$ and $m = 1$

τ	n	$\gamma_0 = 0.05$												$\gamma_0 = 0.1$						$\gamma_0 = 0.2$																																									
		$\theta = 0.02$				$\theta = 0.04$				$\theta = 0.06$				$\theta = 0.08$				$\theta = 0.1$				$\theta = 0.02$				$\theta = 0.04$				$\theta = 0.06$				$\theta = 0.08$				$\theta = 0.1$																							
		$\theta = 0$	$\theta = 0.02$	$\theta = 0.04$	$\theta = 0.06$	$\theta = 0.08$	$\theta = 0.1$	$\theta = 0.02$	$\theta = 0.04$	$\theta = 0.06$	$\theta = 0.08$	$\theta = 0.1$	$\theta = 0.02$	$\theta = 0.04$	$\theta = 0.06$	$\theta = 0.08$	$\theta = 0.1$	$\theta = 0.02$	$\theta = 0.04$	$\theta = 0.06$	$\theta = 0.08$	$\theta = 0.1$	$\theta = 0.02$	$\theta = 0.04$	$\theta = 0.06$	$\theta = 0.08$	$\theta = 0.1$	$\theta = 0.02$	$\theta = 0.04$	$\theta = 0.06$	$\theta = 0.08$	$\theta = 0.1$																													
	$n = 5$	4.31	4.39	4.47	4.56	4.64	4.72	4.30	4.38	4.47	4.55	4.63	4.72	4.35	4.43	4.51	4.60	4.68	4.76	4.30	4.38	4.47	4.55	4.63	4.72	4.30	4.38	4.47	4.55	4.63	4.72	4.30	4.38	4.47	4.55	4.63	4.72	4.30	4.38	4.47	4.55	4.63	4.72	4.30	4.38	4.47	4.55	4.63	4.72	4.30	4.38	4.47	4.55	4.63	4.72						
	$n = 7$	3.10	3.16	3.22	3.29	3.35	3.41	3.07	3.13	3.20	3.26	3.32	3.38	3.11	3.17	3.23	3.29	3.35	3.41	3.07	3.13	3.20	3.26	3.32	3.38	3.07	3.13	3.20	3.26	3.32	3.38	3.07	3.13	3.20	3.26	3.32	3.38	3.07	3.13	3.20	3.26	3.32	3.38	3.07	3.13	3.20	3.26	3.32	3.38	3.07	3.13	3.20	3.26	3.32	3.38	3.07	3.13	3.20	3.26	3.32	3.38
	$n = 10$	2.38	2.41	2.45	2.48	2.52	2.56	2.33	2.37	2.42	2.46	2.49	2.52	2.30	2.34	2.38	2.43	2.47	2.51	2.33	2.37	2.42	2.46	2.49	2.52	2.33	2.37	2.42	2.46	2.49	2.33	2.37	2.42	2.46	2.49	2.52	2.33	2.37	2.42	2.46	2.49	2.52	2.33	2.37	2.42	2.46	2.49	2.52	2.33	2.37	2.42	2.46	2.49	2.52	2.33	2.37	2.42	2.46	2.49	2.52	
	$n = 15$	2.09	2.11	2.12	2.14	2.16	2.18	1.67	1.68	1.72	1.75	1.80	1.83	1.60	1.64	1.70	1.70	1.83	2.176	1.67	1.68	1.72	1.75	1.80	1.83	1.67	1.68	1.72	1.75	1.80	1.67	1.68	1.72	1.75	1.80	1.83	1.67	1.68	1.72	1.75	1.80	1.83	1.67	1.68	1.72	1.75	1.80	1.83	1.67	1.68	1.72	1.75	1.80	1.83							

Table 18.10 The effect of B on the CUSUM- γ^2 charts' performance for $\gamma_0 = 0.05$ (left side), $\gamma_0 = 0.1$ (middle), $\gamma_0 = 0.2$ (right side), $\eta = 0.28, \theta = 0.05$ and $m = 1$

τ	$\gamma_0 = 0.05$						$\gamma_0 = 0.1$						$\gamma_0 = 0.2$					
	$B = 0.8$	$B = 0.9$	$B = 1$	$B = 1.1$	$B = 1.2$	$B = 1.2$	$B = 0.8$	$B = 0.9$	$B = 1$	$B = 1.1$	$B = 1.2$	$B = 1.2$	$B = 0.8$	$B = 0.9$	$B = 1$	$B = 1.1$	$B = 1.2$	
$n = 5$																		
0.5	4.57	4.54	4.51	4.50	4.48	4.48	4.56	4.53	4.51	4.49	4.47	4.47	4.61	4.58	4.55	4.54	4.52	
0.7	8.77	8.71	8.66	8.62	8.58	8.58	8.79	8.72	8.67	8.63	8.59	8.59	8.92	8.85	8.80	8.76	8.72	
0.8	21.4	21.2	21.1	21.0	20.9	20.9	21.5	21.3	21.2	21.1	21.0	21.0	21.9	21.7	21.6	21.4	21.4	
1.3	16.2	16.1	15.9	15.8	15.7	15.7	16.4	16.2	16.0	15.9	15.8	15.8	17.0	16.8	16.6	16.5	16.4	
1.5	6.28	6.20	6.13	6.08	6.03	6.03	6.35	6.27	6.20	6.15	6.10	6.10	6.65	6.56	6.49	6.43	6.38	
2.0	2.64	2.60	2.57	2.55	2.53	2.53	2.68	2.64	2.61	2.58	2.56	2.56	2.83	2.79	2.76	2.73	2.71	
$n = 7$																		
0.5	3.29	3.27	3.26	3.24	3.23	3.23	3.27	3.24	3.23	3.21	3.20	3.20	3.30	3.28	3.26	3.24	3.23	
0.7	6.37	6.32	6.28	6.25	6.22	6.22	6.36	6.31	6.27	6.24	6.21	6.21	6.47	6.41	6.38	6.34	6.32	
0.8	15.9	15.8	15.7	15.6	15.5	15.5	15.9	15.8	15.7	15.6	15.5	15.5	16.3	16.1	16.0	16.0	15.9	
1.3	12.2	12.0	11.9	11.8	11.8	11.8	12.3	12.2	12.0	11.9	11.9	11.9	12.8	12.7	12.5	12.4	12.4	
1.5	4.65	4.59	4.54	4.50	4.47	4.47	4.71	4.65	4.60	4.56	4.53	4.53	4.97	4.90	4.85	4.80	4.77	
2.0	2.00	1.97	1.95	1.93	1.92	1.92	2.03	2.00	1.98	1.96	1.95	1.95	2.16	2.13	2.11	2.09	2.07	
$n = 10$																		
0.5	2.49	2.47	2.46	2.46	2.45	2.45	2.46	2.45	2.44	2.43	2.43	2.43	2.43	2.41	2.41	2.39	2.39	
0.7	4.67	4.64	4.61	4.59	4.57	4.57	4.61	4.58	4.55	4.53	4.51	4.51	4.68	4.64	4.61	4.59	4.57	
0.8	11.8	11.7	11.6	11.5	11.5	11.5	11.7	11.6	11.6	11.5	11.5	11.5	12.0	11.9	11.8	11.8	11.7	
1.3	9.09	8.99	8.90	8.83	8.78	8.78	9.19	9.08	9.00	8.93	8.87	8.87	9.63	9.51	9.42	9.35	9.28	
1.5	3.44	3.40	3.36	3.33	3.31	3.31	3.49	3.45	3.41	3.38	3.36	3.36	3.70	3.65	3.61	3.57	3.55	
2.0	1.55	1.53	1.52	1.50	1.50	1.50	1.57	1.55	1.54	1.53	1.52	1.52	1.67	1.65	1.63	1.62	1.61	
$n = 15$																		
0.5	2.14	2.14	2.13	2.13	2.13	2.13	2.14	2.13	2.13	2.13	2.13	2.13	2.17	2.16	2.16	2.16	2.16	
0.7	3.44	3.40	3.39	3.38	3.36	3.36	3.33	3.31	3.29	3.28	3.27	3.27	3.34	3.31	3.29	3.28	3.26	
0.8	8.59	8.50	8.46	8.41	8.37	8.37	8.46	8.39	8.33	8.28	8.25	8.25	8.62	8.54	8.48	8.43	8.39	
1.3	6.55	6.47	6.41	6.36	6.31	6.31	6.62	6.54	6.48	6.43	6.39	6.39	6.96	6.87	6.81	6.75	6.71	
1.5	2.49	2.46	2.43	2.41	2.40	2.40	2.53	2.49	2.47	2.45	2.43	2.43	2.68	2.64	2.62	2.59	2.57	
2.0	1.23	1.22	1.21	1.21	1.20	1.20	1.24	1.23	1.23	1.22	1.21	1.21	1.31	1.29	1.28	1.28	1.27	

Table 18.11 The effect of m on the CUSUM- γ^2 charts' performance for $\gamma_0 = 0.05$ (left side), $\gamma_0 = 0.1$ (middle), $\gamma_0 = 0.2$ (right side), $\eta = 0.28$, $\theta = 0.05$ and $B = 1$

τ	$\gamma_0 = 0.05$										$\gamma_0 = 0.1$										$\gamma_0 = 0.2$									
	$m = 1$	$m = 3$	$m = 5$	$m = 7$	$m = 10$	$m = 1$	$m = 3$	$m = 5$	$m = 7$	$m = 10$	$m = 1$	$m = 3$	$m = 5$	$m = 7$	$m = 10$	$m = 1$	$m = 3$	$m = 5$	$m = 7$	$m = 10$	$m = 1$	$m = 3$	$m = 5$	$m = 7$	$m = 10$					
$n = 5$																														
0.5	4.51	4.52	4.52	4.52	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.51	4.55	4.55	4.55	4.55	4.55	4.55	4.55	4.55	4.55	4.55	4.55				
0.7	8.66	8.66	8.66	8.66	8.66	8.67	8.67	8.67	8.66	8.66	8.67	8.67	8.67	8.67	8.67	8.80	8.79	8.79	8.79	8.79	8.79	8.80	8.79	8.79	8.79	8.79				
0.8	21.1	21.1	21.1	21.1	21.1	21.2	21.2	21.2	21.1	21.1	21.2	21.2	21.2	21.2	21.2	21.6	21.5	21.5	21.5	21.5	21.5	21.6	21.5	21.5	21.5	21.5				
1.3	15.9	15.9	15.9	15.9	15.9	16.0	16.0	16.0	15.9	15.9	16.0	16.0	16.0	16.0	16.0	16.6	16.6	16.6	16.6	16.6	16.6	16.6	16.6	16.6	16.6	16.6				
1.5	6.13	6.13	6.13	6.13	6.13	6.20	6.20	6.20	6.13	6.13	6.20	6.20	6.19	6.19	6.19	6.49	6.47	6.46	6.46	6.46	6.46	6.49	6.46	6.46	6.46	6.46				
2.0	2.57	2.57	2.57	2.57	2.57	2.61	2.60	2.60	2.57	2.57	2.61	2.60	2.60	2.60	2.60	2.76	2.75	2.75	2.75	2.75	2.75	2.76	2.75	2.75	2.75	2.75				
$n = 7$																														
0.5	3.26	3.26	3.26	3.26	3.26	3.23	3.23	3.23	3.26	3.26	3.23	3.23	3.23	3.23	3.23	3.26	3.25	3.25	3.25	3.25	3.26	3.25	3.25	3.25	3.25	3.25				
0.7	6.28	6.28	6.28	6.28	6.28	6.27	6.27	6.27	6.28	6.28	6.27	6.27	6.27	6.27	6.27	6.38	6.37	6.37	6.37	6.37	6.38	6.37	6.37	6.37	6.37	6.36				
0.8	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	16.0	16.0	16.0	16.0	16.0	16.0	16.0	16.0	16.0	16.0	16.0				
1.3	11.9	11.9	11.9	11.9	11.9	12.0	12.0	12.0	11.9	11.9	12.0	12.0	12.0	12.0	12.0	12.5	12.5	12.5	12.5	12.5	12.5	12.5	12.5	12.5	12.5	12.5				
1.5	4.54	4.54	4.54	4.54	4.54	4.60	4.60	4.60	4.54	4.54	4.60	4.60	4.60	4.60	4.60	4.85	4.83	4.83	4.83	4.83	4.85	4.83	4.83	4.83	4.83	4.82				
2.0	1.95	1.95	1.95	1.95	1.95	1.98	1.98	1.98	1.95	1.95	1.98	1.98	1.98	1.98	1.98	2.11	2.10	2.10	2.10	2.10	2.11	2.10	2.10	2.10	2.10	2.09				
$n = 10$																														
0.5	2.46	2.47	2.47	2.47	2.47	2.44	2.44	2.44	2.47	2.47	2.44	2.44	2.44	2.44	2.44	2.41	2.40	2.41	2.40	2.40	2.41	2.40	2.41	2.40	2.40	2.40				
0.7	4.61	4.61	4.62	4.62	4.62	4.55	4.55	4.55	4.62	4.62	4.55	4.55	4.55	4.55	4.55	4.61	4.61	4.61	4.61	4.61	4.61	4.61	4.61	4.60	4.60	4.60				
0.8	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.6	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8	11.8				
1.3	8.90	8.90	8.90	8.90	8.90	9.00	8.99	8.99	8.90	8.90	9.00	8.99	8.99	8.99	8.99	9.42	9.39	9.39	9.39	9.39	9.42	9.39	9.39	9.38	9.38	9.38				
1.5	3.36	3.36	3.36	3.36	3.36	3.41	3.41	3.41	3.36	3.36	3.41	3.41	3.41	3.41	3.41	3.61	3.59	3.59	3.59	3.59	3.61	3.59	3.59	3.59	3.59	3.59				
2.0	1.52	1.52	1.52	1.52	1.52	1.54	1.54	1.54	1.52	1.52	1.54	1.54	1.54	1.54	1.54	1.63	1.62	1.62	1.62	1.62	1.63	1.62	1.62	1.62	1.62	1.62				
$n = 15$																														
0.5	2.13	2.13	2.13	2.13	2.13	1.79	1.80	1.79	2.13	2.13	1.79	1.79	1.79	1.79	1.79	1.68	1.68	1.68	1.68	1.68	1.68	1.68	1.68	1.68	1.68	1.68				
0.7	3.39	3.40	3.40	3.40	3.40	3.29	3.30	3.29	3.40	3.40	3.29	3.30	3.30	3.30	3.30	3.29	3.29	3.29	3.29	3.29	3.29	3.29	3.29	3.29	3.29	3.29				
0.8	8.46	8.47	8.47	8.47	8.47	8.33	8.33	8.33	8.47	8.47	8.33	8.33	8.33	8.33	8.33	8.48	8.47	8.47	8.47	8.47	8.48	8.47	8.47	8.47	8.47	8.46				
1.3	6.41	6.41	6.41	6.41	6.41	6.48	6.47	6.48	6.41	6.41	6.48	6.47	6.47	6.47	6.47	6.81	6.79	6.78	6.78	6.78	6.81	6.79	6.78	6.78	6.78	6.78				
1.5	2.43	2.43	2.43	2.43	2.43	2.47	2.47	2.47	2.43	2.43	2.47	2.47	2.47	2.46	2.46	2.62	2.61	2.60	2.60	2.60	2.62	2.61	2.60	2.60	2.60	2.60				
2.0	1.21	1.21	1.21	1.21	1.21	1.23	1.22	1.23	1.21	1.21	1.23	1.22	1.22	1.22	1.22	1.28	1.28	1.28	1.28	1.28	1.28	1.28	1.28	1.28	1.28	1.28				

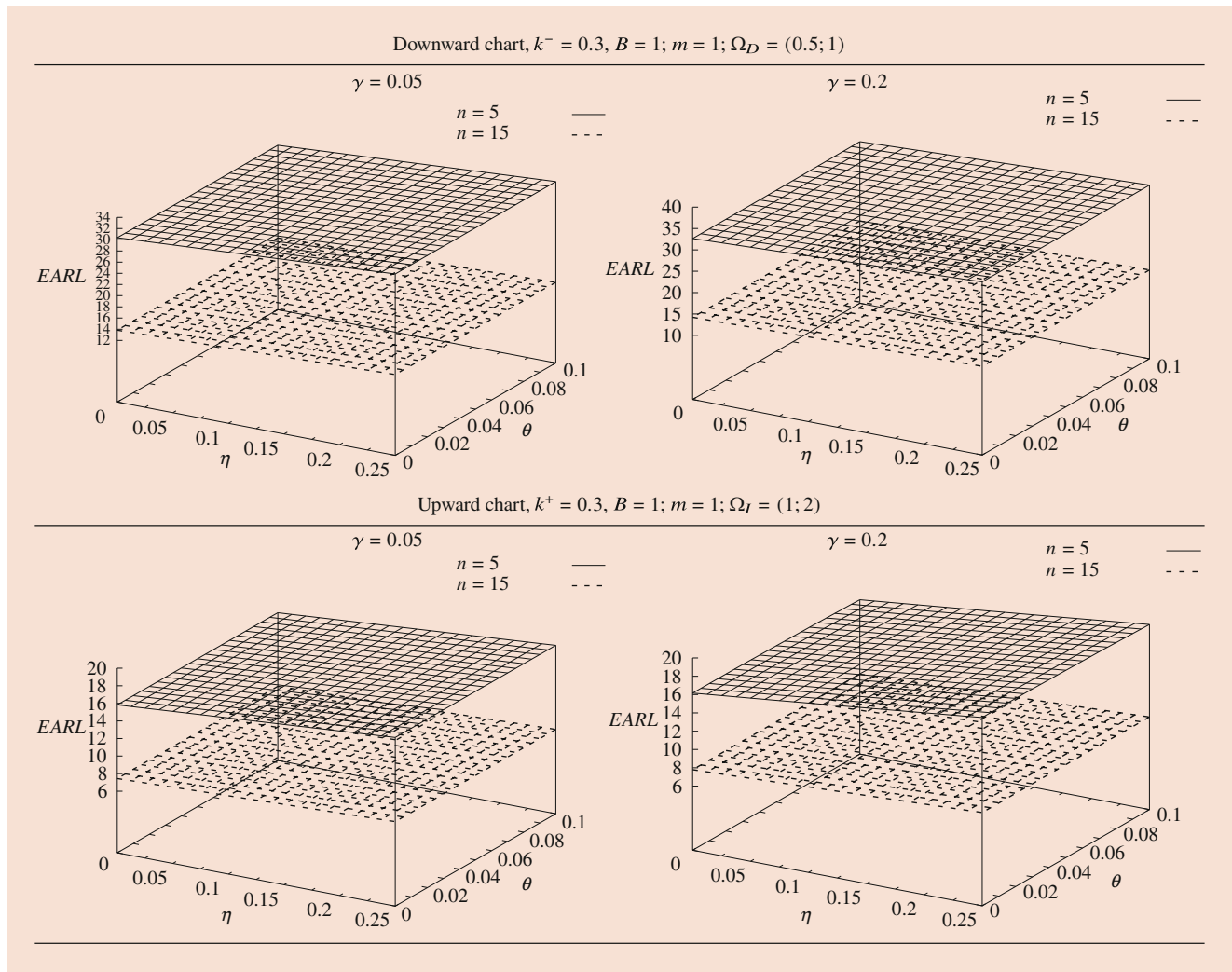


Fig. 18.6 The effect of η and θ on the overall performance of the CUSUM- γ^2 control charts in the presence of measurement error

each value of τ) for $\theta = 0.05, m = 1, B = 1, \eta = \{0, 0.05, 0.1, 0.15, 0.2, 0.25\}, n = \{5, 7\}$ (Table 18.6) and $n = \{10, 15\}$ (Table 18.7). These coefficients for other situations of parameters are not presented here for the sake of brevity but are available upon requests from authors.

Using these optimized coefficients, we can calculate the corresponding out-of-control ARL_1 value. The results are presented in Tables 18.8, 18.9, 18.10, and 18.11. Some conclusions can be drawn from these tables as follows:

- The accuracy error has a negative effect on the performance of the CUSUM- γ^2 control charts: the larger the value of θ , the larger the value of ARL_1 . For example, with $n = 5, \tau = 1.3, \gamma_0 = 0.05, m = 1, B = 1, \eta = 0.28$, we have $ARL_1 = 14.67$ when $\theta = 0$ and $ARL_1 = 17.28$ when $\theta = 0.1$ (see Table 18.9).
- The increase of the linearity error B leads to the decrease of the ARL_1 . For example, from Table 18.10 we have $ARL_1 = 21.52$ for $B = 0.8$ and $ARL_1 = 21.02$ for $B = 1.2$

when $\eta = 0.28, \theta = 0.05, \gamma_0 = 0.1, m = 1$ and $n = 5$. This can be explained by the fact that the rise of B tends to amplify the actual value of observations, making them run out of the control limits faster. However, this does not mean that increasing the linearity error in the measurement system is encouraged because it affects the quality of the measurement, leading to the misunderstanding of the quantity of interest.

- The precision error ratio has not much effect on the performance of the CUSUM- γ^2 control charts. Indeed, when η varies in $[0, 0.25]$, the value of the ARL_1 changes insignificantly as can be seen in Table 18.8.
- The number of multiple measurements per item m has a positive impact on the proposed charts: when m increases, the ARL_1 decreases. However, this impact is not really significant as the decrease of the ARL_1 is trivial when m increases from $m = 1$ to $m = 10$. For example, in Table 18.11 with $\tau = 0.7\eta = 0.028, \theta = 0.05, B = 1, \gamma_0 = 0.2$ and $n = 7$, we have $ARL_1 = 6.29$ when $m = 1$

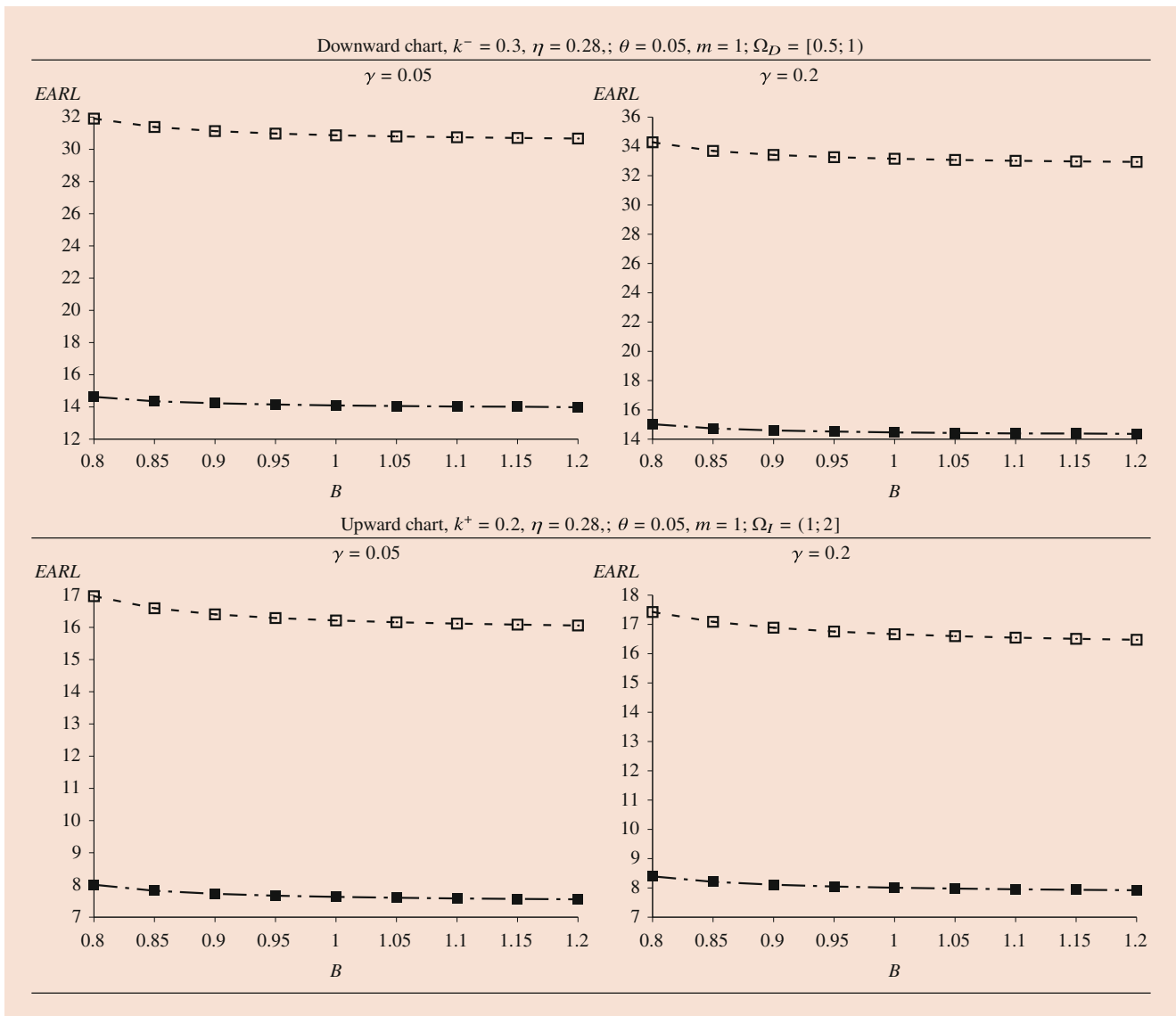


Fig. 18.7 The effect of B on the performance of the CUSUM- γ^2 charts in the presence of measurement error for $n = 5$ ($-\square-$) and $n = 15$ ($-\blacksquare-$)

and $ARL_1 = 6.28$ when $m = 10$. We then conclude that increasing the number of multiple measurement per item is not an efficient way to reduce the effect of ME on the CUSUM- γ^2 control chart.

- As expected, when n increases, the value of the ARL_1 decreases sharply. The increase of n also reduces the negative effect of θ on the CUSUM- γ^2 charts' performance. For example, when $n = 5$, we have $ARL_1 = 19.96$ if $\theta = 0$, compared to $ARL_1 = 22.35$ if $\theta = 0.1$, and when $n = 15$, we have $ARL_1 = 7.93$ if $\theta = 0$ compared to $ARL_1 = 8.99$ if $\theta = 0.1$, where $\tau = 0.8$, $\gamma_0 = 0.05$ (Table 18.9).

The overall effect of ME on the CUSUM- γ^2 control charts when the shift size is not predetermined is shown in Figs. 18.6, 18.7, and 18.8, where the reference coefficient $k^+(k^-)$ is set at 0.3, and the support of τ is $\Omega_I = (1, 2]$ for

the upward control chart, $\Omega_D = [0.5, 1)$ for the downward case. These figures show a similar tendency of the effect of ME as for the case of the specific shift size: the increase of θ reduces significantly the chart performance, the increase of B leads to the decrease of the ARL_1 , the variation of m and η has a trivial impact, and the sample size n has a strong impact on the chart performance.

18.4 Using FIR to Improve the Performance of the CUSUM- γ^2 Control Chart

In the previous design of the CUSUM- γ^2 control chart, we have supposed that $C_0^+ = C_0^- = 0$, i.e., the process starts from an in-control state. However, it is not always the case

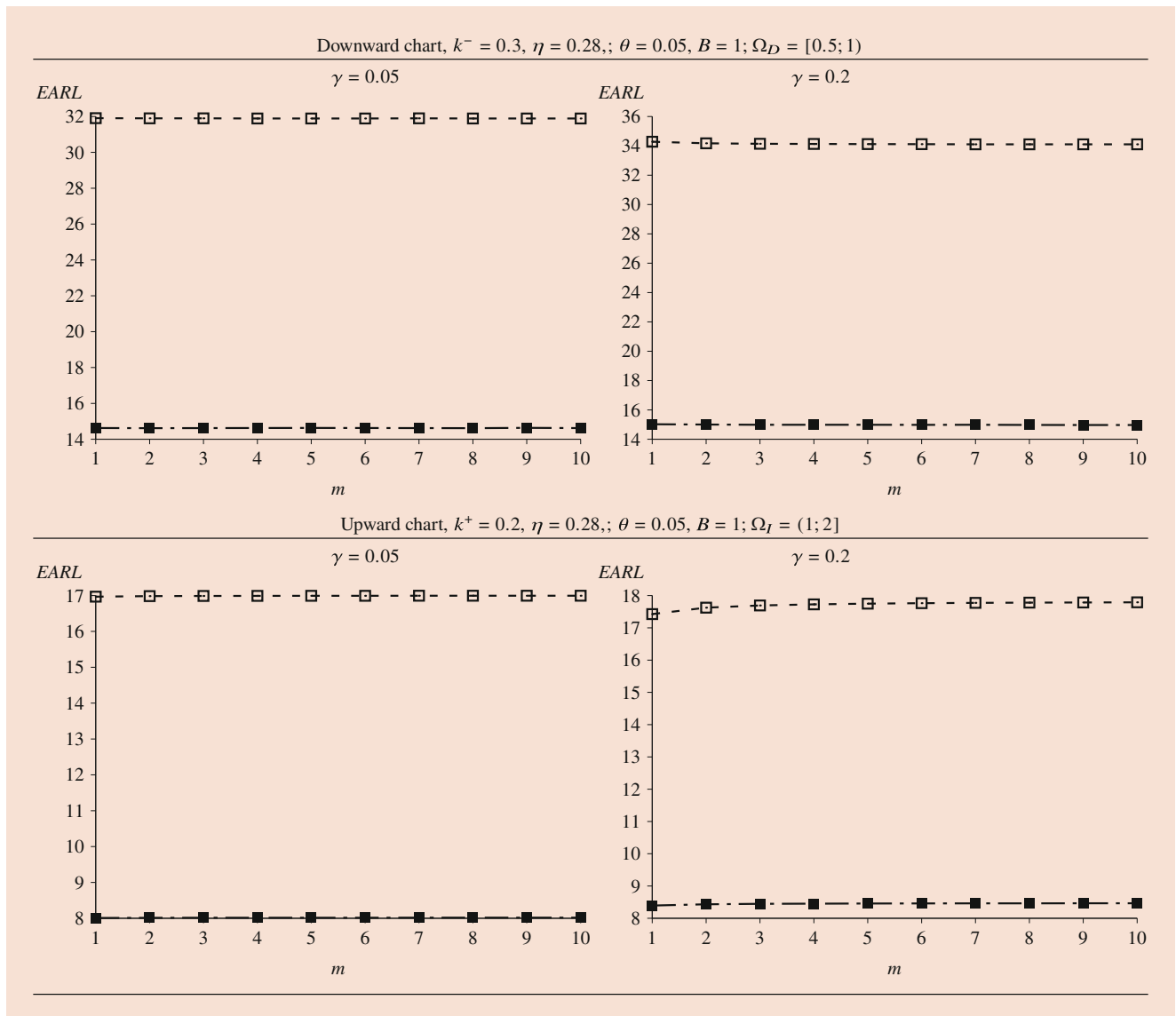


Fig. 18.8 The effect of m on the performance of the CUSUM- γ^2 charts in the presence of measurement error for $n = 5$ ($-\square-$) and $n = 15$ ($-\blacksquare-$)

as the process may be in the out-of-control state at startup or when the process is restarted after an adjustment has been made to the process in practice. In order to improve the performance of the CUSUM chart in such special situations, Lucas and Crosier [64] suggested a fast initial response (FIR) strategy (FIR) strategy. According to this method, the standard CUSUM procedure is modified by initially setting a CUSUM to a specified positive *headstart* $C_0^+ = C_0^- = c$. The design of the FIR CUSUM chart is still the same as the design of the standard CUSUM chart. However, instead of setting up $c = 0$, in the FIR CUSUM chart, we choose the value of c between 0 and the control limit H .

The FIR strategy has a valuable advantage in that it enhances significantly the CUSUM chart performance when the process starts from an out-of-control state but costs only a

small penalty if the process starts in control. The price for this improvement and for maintaining the same in-control ARL_0 is the somewhat increase of the control limit H . In the literature, the FIR was used to improve the performance of the CUSUM chart monitoring process dispersion in a study by Sanusi et al. [65]. An extensive discussion of the FIR feature can be seen in a study by Hawkins and Olwell [7].

In this section, we apply the FIR strategy in designing the CUSUM- γ^2 control chart. The value of optimal reference coefficients h^{*+} (h^{*-}) and the corresponding ARL_1 of the FIR CUSUM- γ^2 are presented in Tables 18.12 and 18.13, respectively. In the current study, the FIR CUSUM- γ^2 control chart is applied with the values of k^{*+} and k^{*-} in Tables 18.2 and 18.3. We consider several situations of the headstart, that is, $c \in \{0, 0.25H, 0.5H, 0.75H\}$. It can be seen that the use

Table 18.12 The chart parameters h^{*+} when $\tau = \{0.5, 0.7, 0.8\}$, h^{*-} when $\tau = \{1.3, 1.5, 2.0\}$ of the CUSUM- γ^2 charts for $n = \{5, 7, 10, 15\}$, $\gamma_0 = \{0.05, 0.1, 0.2\}$, and different values of the head start c

τ	$\gamma_0 = 0.05$						$\gamma_0 = 0.1$						$\gamma_0 = 0.2$								
	$C_0 = 0$		$C_0 = 0.1H$		$C_0 = 0.25H$		$C_0 = 0.5H$		$C_0 = 0.75H$		$C_0 = 0$		$C_0 = 0.1H$		$C_0 = 0.25H$		$C_0 = 0.5H$		$C_0 = 0.75H$		
$n = 5$																					
0.5	0.8105	0.8108	0.8119	0.8168	0.8351	0.7915	0.7918	0.7929	0.7975	0.8149	0.8127	0.8130	0.8141	0.8188							
0.7	1.6283	1.6291	1.6323	1.6474	1.6994	1.6444	1.6452	1.6484	1.6635	1.7161	1.7126	1.7135	1.7168	1.7325							
0.8	3.3713	3.3745	3.3867	3.4426	3.6154	3.3938	3.3970	3.4092	3.4648	3.6381	3.5586	3.5619	3.5747	3.6330							
1.3	5.9685	5.9728	5.9924	6.0875	6.3666	4.9781	4.9781	4.9781	4.9781	4.9781	5.6251	5.6251	5.6251	5.6251							
1.5	4.2236	4.2254	4.2321	4.2644	4.3623	3.6291	3.6291	3.6291	3.6291	3.6291	4.1954	4.1954	4.1954	4.1954							
2.0	3.0593	3.0602	3.0632	3.0750	3.1082	2.6300	2.6300	2.6300	2.6300	2.6300	3.1351	3.1351	3.1351	3.1351							
$n = 7$																					
0.5	0.5307	0.5309	0.5316	0.5351	0.5429	0.5274	0.5275	0.5282	0.5315	0.5392	0.5421	0.5423	0.5429	0.5463							
0.7	1.1324	1.1329	1.1348	1.1434	1.1740	1.1400	1.1406	1.1424	1.1509	1.1819	1.1631	1.1636	1.1655	1.1740							
0.8	2.3581	2.3599	2.3667	2.3984	2.4982	2.3851	2.3868	2.3937	2.4255	2.5256	2.5021	2.5040	2.5112	2.5449							
1.3	4.0796	4.0821	4.0925	4.1432	4.2962	3.5308	3.5308	3.5308	3.5308	3.5308	3.9375	3.9375	3.9375	3.9375							
1.5	2.8258	2.8269	2.8306	2.8478	2.8992	2.4893	2.4893	2.4893	2.4893	2.4893	2.8450	2.8450	2.8450	2.8450							
2.0	1.9521	1.9526	1.9542	1.9602	1.9754	1.7075	1.7075	1.7075	1.7075	1.7075	2.0195	2.0195	2.0195	2.0195							
$n = 10$																					
0.5	0.5557	0.5562	0.5568	0.5606	0.5687	0.3581	0.3582	0.3586	0.3602	0.3633	0.3554	0.3555	0.3559	0.3573							
0.7	0.7932	0.7935	0.7949	0.8002	0.8157	0.7884	0.7888	0.7899	0.7950	0.8108	0.7917	0.7921	0.7931	0.7982							
0.8	1.7024	1.7035	1.7077	1.7273	1.7884	1.6708	1.6719	1.6758	1.6942	1.7541	1.7415	1.7426	1.7467	1.7657							
1.3	2.7817	2.7831	2.7888	2.8161	2.9000	2.4735	2.4735	2.4735	2.4735	2.4735	2.7378	2.7378	2.7378	2.7378							
1.5	1.8720	1.8726	1.8747	1.8837	1.9098	1.6786	1.6786	1.6786	1.6786	1.6786	1.9056	1.9056	1.9056	1.9056							
2.0	1.1979	1.1982	1.1991	1.2019	1.2082	1.0606	1.0606	1.0606	1.0606	1.0606	1.2547	1.2547	1.2547	1.2547							
$n = 15$																					
0.5	0.6833	0.6840	0.6849	0.6900	0.7029	0.3121	0.3121	0.3124	0.3136	0.3159	0.2973	0.2974	0.2977	0.2986							
0.7	0.6833	0.6840	0.6849	0.6900	0.7029	0.6981	0.6985	0.6994	0.7042	0.7179	0.5143	0.5145	0.5151	0.5183							
0.8	1.1938	1.1945	1.1971	1.2091	1.2442	1.1810	1.1817	1.1841	1.1952	1.2311	1.1855	1.1861	1.1884	1.1992							
1.3	1.8224	1.8232	1.8262	1.8402	1.8832	1.6566	1.6566	1.6566	1.6566	1.6566	1.8239	1.8239	1.8239	1.8239							
1.5	1.1747	1.1750	1.1761	1.1805	1.1921	1.0689	1.0689	1.0689	1.0689	1.0689	1.2060	1.2060	1.2060	1.2060							
2.0	0.6405	0.6406	0.6410	0.6421	0.6441	0.5695	0.5695	0.5695	0.5695	0.5695	0.6898	0.6898	0.6898	0.6898							

Table 18.13 The ARL_1 values of the $CUSUM_{-\gamma^2}$ control charts for $n = \{5, 7, 10, 15\}$, $\gamma_0 = \{0.05, 0.1, 0.2\}$, and different values of the head start c

τ	$\gamma_0 = 0.2$											
	$\gamma_0 = 0.1$				$\gamma_0 = 0.1$				$\gamma_0 = 0.1$			
	$C_0 = 0$	$C_0 = 0.1H$	$C_0 = 0.5H$	$C_0 = 0.75H$	$C_0 = 0$	$C_0 = 0.1H$	$C_0 = 0.5H$	$C_0 = 0.75H$	$C_0 = 0$	$C_0 = 0.1H$	$C_0 = 0.5H$	$C_0 = 0.75H$
$n = 5$												
0.5	4.3079	4.0416	3.5052	2.6804	4.3003	4.0345	3.5043	2.6776	1.6772	4.3456	4.0784	3.5421
0.7	8.1989	7.7462	6.7763	4.9751	8.2106	7.7572	6.7855	4.9817	3.0706	8.3359	7.8771	6.8912
0.8	19.962	19.085	16.928	12.498	20.025	19.149	16.992	12.555	7.6025	20.397	19.510	17.326
1.3	18.091	17.552	15.973	12.336	14.795	14.362	13.149	10.267	6.8895	15.351	14.909	13.641
1.5	6.4474	6.2173	5.6783	4.5463	5.6136	5.4229	4.9876	4.0679	3.0505	5.8872	5.6906	5.2320
2.0	2.5488	2.4682	2.3062	2.0105	2.3424	2.2760	2.1435	1.9036	1.6523	2.4857	2.4145	2.2698
$n = 7$												
0.5	3.1006	2.9333	2.6218	1.9139	3.0740	2.9104	2.6007	1.8967	1.3736	3.1024	2.9353	2.6262
0.7	5.9432	5.6091	4.9077	3.6746	5.9321	5.5986	4.8980	3.6680	2.2830	6.0297	5.6933	4.9851
0.8	14.800	14.118	12.500	9.2497	14.834	14.149	12.527	9.2676	5.7046	15.155	14.457	12.803
1.3	12.956	12.519	11.359	8.7826	11.088	10.720	9.7721	7.6264	5.1789	11.576	11.198	10.204
1.5	4.6381	4.4592	4.0692	3.3018	4.1667	4.0136	3.6849	3.0400	2.3533	4.3996	4.2397	3.8906
2.0	1.9006	1.8462	1.7418	1.5653	1.7959	1.7503	1.6631	1.5161	1.3734	1.9117	1.8608	1.7618
$n = 10$												
0.5	2.3814	2.2766	2.0160	1.4281	2.3258	2.1787	1.8805	1.4515	1.1937	2.2976	2.1498	1.8563
0.7	4.3608	4.1065	3.6087	2.7167	4.2995	4.0487	3.5620	2.6781	1.7704	4.3602	4.1081	3.6183
0.8	10.941	10.403	9.1791	6.7932	10.919	10.388	9.1791	6.8088	4.2264	11.171	10.630	9.3950
1.3	9.3038	8.9547	8.1020	6.2910	8.2461	7.9417	7.2120	5.6479	3.9130	8.6491	8.3337	7.5660
1.5	3.3568	3.2225	2.9461	2.4387	3.0931	2.9758	2.7365	2.2998	1.8590	3.2794	3.1550	2.8977
2.0	1.4621	1.4305	1.3725	1.2809	1.4156	1.3889	1.3399	1.2623	1.1919	1.4995	1.4678	1.4087
$n = 15$												
0.5	2.0939	2.0444	1.8546	1.1956	1.6885	1.5623	1.3555	1.1366	1.0434	1.6301	1.5138	1.3278
0.7	3.2237	3.0295	2.6799	1.9900	3.1449	2.9562	2.6169	1.9369	1.3432	3.1235	2.9569	2.6047
0.8	7.9487	7.5346	6.6324	4.9397	7.8287	7.4208	6.5330	4.8698	3.0579	7.9730	7.5046	6.6716
1.3	6.5050	6.2372	5.6301	4.4103	5.9320	5.6919	5.1539	4.0703	2.9129	6.2438	5.9932	5.4248
1.5	2.3839	2.2914	2.1113	1.8054	2.2475	2.1658	2.0075	1.7401	1.4897	2.3860	2.2975	2.1240
2.0	1.1719	1.1596	1.1379	1.1053	1.1583	1.1479	1.1293	1.1013	1.0771	1.2069	1.1928	1.1677

of FIR leads to the better performance of the CUSUM- γ^2 control chart. In particular, when $C_0 = c$ increases, the ARL_1 decreases. For example, with $n = 5$, $\gamma_0 = 0.1$, $\tau = 0.8$, we have $ARL_1 = 20.025$ when $C_0^- = 0$ and $ARL_1 = 7.6025$ when $C_0^- = 0.75H^-$. In addition, the increase of c increases the value of $h^{*+}(h^{*-})$ (corresponding to the control limit $H^{*+}(H^{*-})$) as well. For example, when $n = 7$, $\gamma_0 = 0.05$, $\tau = 1.3$, the optimal values of reference coefficient h^{*+} is $h^{*+} = 4.0796$ if $C_0 = 0$ and $h^{*+} = 4.2962$ if $C_0 = 0.75H^+$. These results are consistent with the findings by Lucas and Crosier [64]. Using a head start routinely is then recommended as a good strategy to obtain better performance in designing a CUSUM control chart monitoring the CV.

18.5 Conclusions and Perspectives

In this chapter, we have investigated several problems related to the CUSUM control chart monitoring the CV, involving the design of the CUSUM- γ^2 control chart, the effect of ME, and the FIR strategy to improve the performance of the proposed charts. The chart performance is measured by using the ARL , which is the expected number of samples to be taken before a signal from the chart. This measurement has been calculated based on an approximate Markov chain method. We have also considered the measurement of $EARL$ when a specific shift size of the process cannot be predicted accurately in advance.

Thanks to the definition of the CUSUM statistic, the CUSUM- γ^2 control chart is able to cumulate information on the whole process from the past. As a result, it is quite sensitive to the process shift and it outperforms the statistical performance of the traditional Shewhart control chart.

The numerical results have shown that the ME has a significant impact on the chart's performance. In particular, the increase of the accuracy error or the precision error leads to the decrease of the efficiency of the chart in detecting process shifts. Moreover, the traditional method of increasing the number of multiple measurements per item is not an efficient way to reduce these negative effects of ME. By contrast, increasing the sample size is a very effective way to enhance the performance of the CUSUM- γ^2 control chart regardless of ME.

Another way to improve the CUSUM- γ^2 control chart's performance is to apply the FIR strategy. In this strategy, the process is supposed to start from an out-of-control state. Then, the initial values of the CUSUM statistics are chosen to be strictly greater than 0. This choice could lead to somewhat larger interval controls, but it can reduce significantly the average number of samples that need to be taken from the chart to alarm an abnormal state. These findings could be useful for practitioners in designing a CUSUM control chart to monitor the CV.

The obtained results in this chapter can be applied in various industrial environments when the change in the CV is considered as a source that leads to the change of the process output and the small and moderate shifts in the CV are expected to detect as quickly as possible. It is also applied for the case when one considers a practical problem of ME in the implementation of the CUSUM- γ^2 chart.

From this chapter, several important problems related to the CUSUM- γ^2 control chart could be investigated in future research. For example, one can think of combining the CUSUM chart with other adaptive strategies like changing the sample size, changing the sampling interval, or changing both of them to obtain a control chart monitoring the CV (i.e., the VSI CUSUM- γ^2 chart, VSS CUSUM- γ^2 chart, or VSSI CUSUM- γ^2 chart) with higher performance. Moreover, in Sect. 18.3, we have modeled ME by applying a linear co-variate error model. In some situations, the additive linear model could be too simple and it does not model ME well. For such situations, one can think about using a more general model for ME like the two-component error model suggested by Rocke and Lorenzato [66]. However, the general model is such that the observations are no longer normal and simulation is needed to get the control charts' statistical properties. Finally, investigating the effect of skew distributions on the CUSUM- γ^2 is also an interesting problem with a practical meaning. In the SPC literature, this problem has been presented in several studies (see, e.g., Bai and Choi [67], Derya and Canan [68]). Particularly, Stoumbos and Reynolds [1] also studied the robustness and performance of CUSUM control charts based on a double-exponential distribution. However, a CUSUM control chart monitoring the CV with a skew distribution has not been investigated yet, and it is worth considering it to make it more practical for practitioners.

Acknowledgments Research activities of Phuong Hanh Tran have been supported by Univ. Lille, ENSAIT, ULR 2461 – GEMTEX – Engineering and Textile Materials, Roubaix, France and HEC Liège-Management School of the University of Liège, Liège, Belgium under grant 2020/MOB/00504.

References

1. Stoumbos, Z.G., Reynolds, M.R.: The Robustness and Performance of CUSUM Control Charts Based on the Double-Exponential and Normal Distributions. In: Lenz, H.J., Wilrich, P.T. (eds.) *Frontiers in Statistical Quality Control*, vol. 7. Physica, Heidelberg (2004)
2. Castagliola, P., Celano, G., Fichera, S.: Monitoring process variability using EWMA. In: Pham, H. (ed.) *Handbook of Engineering Statistics*. Springer, Berlin (2006)
3. Chang, W.R., McLean, I.P.: CUSUM: a tool for early feedback about performance? *BMC Med. Res. Methodol.* **6**(1), 1–8 (2006)
4. Yashchin, E.: Performance of CUSUM control schemes for serially correlated observations. *Technometrics* **35**(1), 37–52 (1993)
5. Saghaei, A., Mehrjoo, M., Amiri, A.: A CUSUM-based method for monitoring simple linear profiles. *Int. J. Adv. Manuf. Technol.* **45**, 1252–1260 (2009)

6. Castagliola, P., Maravelakis, P.E.: A CUSUM control chart for monitoring the variance when parameters are estimated. *J. Stat. Plan. Inference*. **141**, 1463–1478 (2011)
7. Hawkins, D.M., Olwell, D.H.: *Cumulative Sum Charts and Charting for Quality Improvement*. Springer Science & Business Media, Berlin (2012)
8. Castagliola, P., Celano, G., Psarakis, S.: Monitoring the coefficient of variation using EWMA charts. *J. Qual. Technol.* **43**(3), 249–265 (2011)
9. Yeong, W.C., Khoo, M.B.C., Teoh, W.L., Castagliola, P.: A control chart for the multivariate coefficient of variation. *Qual. Reliab. Eng. Int.* **32**(3), 1213–1225 (2016)
10. Ye, J., Feng, P., Xu, C., Ma, Y., Huang, S.: A novel approach for chatter online monitoring using coefficient of variation in machining process. *Int. J. Adv. Manuf. Technol.* **96**(1–4), 287–297 (2018)
11. Muhammad, A.N.B., Yeong, W.C., Chong, Z.L., Lim, S.L., Khoo, M.B.C.: Monitoring the coefficient of variation using a variable sample size EWMA chart. *Comput. Ind. Eng.* **126**, 378–398 (2018)
12. Kang, C.W., Lee, M.S., Seong, Y.J., Hawkins, D.M.: A control chart for the coefficient of variation. *J. Qual. Technol.* **39**(2), 151–158 (2007)
13. Hong, E.-P., Chang-Wook, K., Jae-Won, B., Hae-Woon, K.: Development of CV control chart using EWMA technique. *J. Soc. Korea Ind. Syst. Eng.* **31**(4), 114–120 (2008)
14. Castagliola, P., Achouri, A., Taleb, H., Celano, G., Psarakis, S.: Monitoring the coefficient of variation using control charts with Run Rules. *Qual. Technol. Quant. Manag.* **10**, 75–94 (2013)
15. Hanh, T.P., Cédric, H., Du, N.H., Hélène, M.: Monitoring coefficient of variation using one-sided run rules control charts in the presence of measurement errors. *J. Appl. Stat.* **12**(48), 2178–2204 (2021)
16. Calzada, M.E., Scariano, S.M.: A synthetic control chart for the coefficient of variation. *J. Stat. Comput. Simul.* **83**(5), 853–867 (2013)
17. Teoh, W.L., Khoo, M.B.C., Yeong, W.C., Teh, S.Y.: Comparison between the performances of synthetic and EWMA charts for monitoring the coefficient of variation. *J. Scient. Res. Develop.* **3**(1), 16–20 (2016)
18. Zhang, J., Li, Z., Chen, B., Wang, Z.: A new exponentially weighted moving average control chart for monitoring the coefficient of variation. *Comput. Ind. Eng.* **78**, 205–212 (2014)
19. Hong, E.-P., Kang, H.W., Kang, C.W., Baik, J.W.: CV control chart using GWMA technique. *Adv. Mat. Res.* **337**, 247–254 (2011)
20. Lin, T.W., Khoo, M.B.C., Castagliola, P., Yeong, W.C., Yin, T.S.: Run-sum control charts for monitoring the coefficient of variation. *Qual. Reliab. Eng. Int.* **257**(1), 144–158 (2017)
21. Hanh, T.P., Phuc, T.K.: The efficiency of CUSUM schemes for monitoring the coefficient of variation. *Appl. Stoch. Models Bus. Ind.* **32**(6), 870–881 (2016)
22. Hanh, T.P., Cédric, H., Sébastien, T.: Enhanced CUSUM control charts for monitoring coefficient of variation: A case study in Textile industry. *IFAC-PapersOnLine*, **55**(10), 1195–1200 (2022)
23. Castagliola, P., Achouri, A., Taleb, H., Celano, G., Psarakis, S.: Monitoring the coefficient of variation using a variable sampling interval control chart. *Qual. Reliab. Eng. Int.* **29**(8), 1135–1149 (2013)
24. Castagliola, P., Achouri, A., Taleb, H., Celano, G., Psarakis, S.: Monitoring the coefficient of variation using a variable sample size control chart. *Int. J. Adv. Manuf. Technol.* **81**(9–12), 1561–1576 (2015)
25. Yeong, W.C., Khoo, M.B.C., Lim, S.L., Lee, M.H.: A direct procedure for monitoring the coefficient of variation using a variable sample size scheme. *Commun. Stat. Simul. Comput.* **46**(6), 4210–4225 (2017)
26. Khaw, K.W., Khoo, M.B.C., Yeong, W.C., Wu, Z.: Monitoring the coefficient of variation using a variable sample size and sampling interval control chart. *Commun. Stat. Simul. Comput.* **46**(7), 5722–5794 (2017)
27. Yeong, W.C., Lim, S.L., Khoo, M.B.C., Castagliola, P.: Monitoring the coefficient of variation using a variable parameters chart. *Qual. Eng.* **30**(2), 212–235 (2018)
28. Yeong, W.C., Khoo, M.B.C., Teoh, W.L., Abdur, R.: Monitoring the coefficient of variation using a variable sampling interval EWMA chart. *J. Qual. Technol.* **49**(3), 380–401 (2017)
29. Hanh, T.P., Cédric, H.: Monitoring the coefficient of variation using variable sampling interval CUSUM control charts. *J. Stat. Comput. Simul.* **3**(91), 501–521 (2021)
30. Castagliola, P., Amdouni, A., Tale, H., Celano, G.: One-sided Shewhart-type charts for monitoring the coefficient of variation in short production runs. *Qual. Technol. Quant. Manag.* **12**(1), 53–67 (2015)
31. Amdouni, A., Castagliola, P., Tale, H., Celano, G.: A variable sampling interval Shewhart control chart for monitoring the coefficient of variation in short production runs. *Int. J. Prod. Res.* **55**(19), 5521–5536 (2017)
32. Amdouni, A., Castagliola, P., Tale, H., Celano, G.: One-sided run rules control charts for monitoring the coefficient of variation in short production runs. *Euro. J. Ind. Eng.* **10**, 639–663 (2016)
33. Amdouni, A., Castagliola, P., Tale, H., Celano, G.: Monitoring the coefficient of variation using a variable sample size control chart in short production runs. *Int. J. Adv. Manuf. Technol.* **81**, 1–14 (2015)
34. Dawod, A.B.A., Abbasi, S.A., Al-Momani, M.: On the performance of coefficient of variation control charts in Phase I. *Qual. Reliab. Eng. Int.* **34**(6), 1029–1040 (2018)
35. Yeong, W.C., Khoo, M.B.C., Lim, S.L., Teoh, W.L.: The coefficient of variation chart with measurement error. *Qual. Technol. Quant. Manag.* **14**(4), 353–377 (2017)
36. Phuc, T.K., Du, N.H., Thong, N.Q., Chattinnawat, W.: One-sided synthetic control charts for monitoring the coefficient of variation with measurement errors. In: 2018 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 1667–1671 (2018)
37. Du, N.H., Thong, N.Q., Phuc, T.K., Phuc, H.D.: On the performance of VSI Shewhart control chart for monitoring the coefficient of variation in the presence of measurement errors. *Int. J. Adv. Manuf. Technol.* (2019). <https://doi.org/10.1007/s00170-019-03352-7>
38. Phuc, T.K., Du, N.H., Hanh, T.P., Cedric, H.: On the performance of CUSUM control charts for monitoring the coefficient of variation with measurement errors. *Int. J. Adv. Manuf. Technol.* (2019) <https://doi.org/10.1007/s00170-019-03987-6>
39. Montgomery, D.C.: *Introduction to Statistical Quality Control*, 7th edn. John Wiley & Sons, Hoboken (2009)
40. Hawkins, D.M., Olwell, D.H.: *Cumulative Sum Charts and Charting for Quality Improvement*, 1st edn. Springer, New York (1998)
41. Woodall, W., Adams, B.: The statistical design of CUSUM charts. *Qual. Eng.* **5**(4), 559–570 (1993)
42. Hawkins, D.M.: A fast accurate approximation for average run lengths of CUSUM control charts. *J. Qual. Technol.* **24**(1), 37–43 (1992)
43. Page, E.S.: Control charts with warning lines. *Biometrics* **42**, 243–257 (1955)
44. Crowder, S.V.: A simple method for studying run-length distributions of exponentially weighted moving average charts. *Technometrics* **29**(4), 401–407 (1987)
45. Brook, D., Evans, D.: An approach to the probability distribution of CUSUM run length. *Biometrika* **69**(3), 539–549 (1972)

46. Neuts, M.F.: Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach. Johns Hopkins University Press, Baltimore (1981)
47. Latouche, G., Ramaswami, V.: Introduction to matrix analytic methods in stochastic modelling. Series on Statistics and Applied Probability. SIAM, Philadelphia (1999)
48. McKay, A.T.: Distribution of the coefficient of variation and extended t distribution. *J. R. Stat. Soc.* **95**(4), 695–698 (1932)
49. Hendricks, W.A., Robey, K.W.: The sampling distribution of the coefficient of variation. *Ann. Math. Stat.* **7**(3), 129–132 (1936)
50. Iglewicz, B., Myers, R.H.: Comparisons of approximations to the percentage points of the sample coefficient of variation. *Technometrics* **12**(1), 166–169 (1970)
51. Iglewicz, B., Myers, R.H., Howe, R.B.: On the percentage points of the sample coefficient of variation. *Biometrika* **55**(3), 580–581 (1968)
52. Vangel, M.G.: Confidence intervals for a normal coefficient of variation. *Am. Stat.* **15**, 21–26 (1996)
53. Warren, W.G.: On the adequacy of the chi-squared approximation for the coefficient of variation. *Commun. Stat. Simul. Comput.* **11**, 659–666 (1982)
54. Reh, W., Scheffler, B.: Significance tests and confidence intervals for coefficients of variation. *Comput. Stat. Data Anal.* **22**(4), 449–452 (1996)
55. Breunig, R.: An almost unbiased estimator of the coefficient of variation. *Eco. Lett.* **70**(1), 15–19 (2001)
56. Bennet, C.A.: Effect of measurement error on chemical process control. *Ind. Qual. Control* **10**(4), 17–20 (1954)
57. Linna, K.W., Woodall, W.H.: Effect of measurement error on Shewhart control chart. *J. Qual. Technol.* **33**(2), 213–222 (2001)
58. Maravelakis, P.E.: Measurement error effect on the CUSUM control chart. *J. Appl. Stat.* **39**(2), 323–336 (2012)
59. Maravelakis, P., Panaretos, J., Psarakis, S.: EWMA chart and measurement error. *J. Appl. Stat.* **31**(4), 445–455 (2004)
60. Noorossana, R., Zerehsaz, Y.: Effect of measurement error on phase II monitoring of simple linear profiles. *Int. J. Adv. Manuf. Technol.* **79**(9–12), 2031–2040 (2015)
61. Hanh, T.P., Phuc, T.K., Rakitzis, A.: A synthetic median control chart for monitoring the process mean with measurement errors. *Qual. Reliab. Eng. Int.* **35**(4), 1100–1116 (2019)
62. Phuc, T.K., Cedric, H., Balakrishnan, N.: On the performance of coefficient of variation charts in the presence of measurement errors. *Qual. Reliab. Eng. Int.* **35**(1), 329–350 (2018)
63. Down, M., Czubak, F., Gruska, G., Stahley, S., Benham, D.: Measurement System Analysis, 4th edn. Chrysler Group LLC, Ford Motor Company, General Motors Corporation, Auburn Hills (2010)
64. Lucas, J.M., Crosier, R.B.: Fast initial response for CUSUM quality-control schemes: give your CUSUM a head start. *Technometrics* **24**(3), 199–205 (1982)
65. Sanusi, R.A., Riaz, M., Abbas, N., Abujiya, M.R.: Using FIR to improve CUSUM charts for monitoring process dispersion. *Qual. Reliab. Eng. Int.* **33**(5), 1045–1056 (2017)
66. Rocke, D.M., Lorenzato, S.: A two-Component model for measurement error in analytical chemistry. *Technometrics* **37**(2), 176–184 (1995)
67. Bai, D.S., Choi, I.: \bar{X} and R control charts for skewed populations. *J. Qual. Technol.* **27**(2), 120–131 (1995)
68. Derya, K., Canan, H.: Control charts for Skewed distributions: Weibull, Gamma, and Lognormal. *Metodoloski zvezki* **9**(2), 95–106 (2012)



Phuong Hanh TRAN is currently a Ph.D student and researcher at HEC Liège - Management School of the University of Liège and QuantOM (Center of Quantitative Methods and Operations Management), Liège, Belgium. She is also a visiting researcher at the University of Lille, ENSAIT & GEMTEX, Lille, France. Her research is now focused on developing new Statistical Process Control (SPC) and Machine Learning methods for anomaly detection in Business, Finance and Industry Management.



Kim Phuc Tran is currently a Senior Associate Professor (Maître de Conférences HDR, equivalent to a UK Reader) of Artificial Intelligence and Data Science at the ENSAIT and the GEMTEX laboratory, University of Lille, France. He received an Engineer's degree and a Master of Engineering degree in Automated Manufacturing. He obtained a Ph.D. in Automation and Applied Informatics at the University of Nantes, and an HDR (Doctor of Science or Dr. habil.) in Computer Science and Automation at the University of Lille, France. His research deals with Real-time Anomaly Detection with Machine Learning with applications, Decision Support Systems with Artificial Intelligence, and Enabling Smart Manufacturing with IIoT, Federated learning, and Edge computing. He has published more than 64 papers in peer-reviewed international journals and proceedings at international conferences. He edited 3 books with Springer Nature and Taylor & Francis. He is the Associate Editor, Editorial Board Member, and Guest Editor for several international journals such as IEEE Transactions on Intelligent Transportation Systems and Engineering Applications of Artificial Intelligence. He has supervised 9 Ph.D. students and 3 Postdocs. In addition, as the project coordinator (PI), he is conducting 1 regional research project about Healthcare Systems with Federated Learning. He has been or is involved in 8 regional, national, and European projects. He is an expert and evaluator for the Public Service of Wallonia (SPW-EER), Belgium, and the Natural Sciences and Engineering Research Council of Canada. He received the Award for Scientific Excellence (Prime d'Encadrement Doctoral et de Recherche) given by the Ministry of Higher Education, Research and Innovation, France for 4 years from 2021 to 2025 in recognition of his outstanding scientific achievements. From 2017 until now, he has been the Senior Scientific Advisor at Dong A University and the International Research Institute for Artificial Intelligence and Data Science (IAD), Danang, Vietnam where he has held the International Chair in Data Science and Explainable Artificial Intelligence.



Prof. Cédric Heuchenne is a Full Professor of Statistics and Quantitative Methods at HEC Liège - Management School of the University of Liège, and also teaches at the Catholic University of Louvain. More precisely, his research concerns different applied or more theoretical fields like survival or duration data analysis, nonparametric statistical inference, statistical process control, quality management, or risk modelling.



Dr. Huu Du Nguyen is a member of the International Research Institute for Artificial Intelligence and Data Science, Dong A University, Danang, Viet Nam. He is a member of the Faculty of Information Technology, National University of Agriculture, Hanoi, Vietnam. His research is related to reliability assessment, statistical process control, machine learning and deep learning.



Change-Point-Based Statistical Process Controls

19

Partha Sarathi Mukherjee

Contents

19.1	Introduction	361
19.2	Traditionally Used SPC Charts	363
19.2.1	Shewhart Chart.....	363
19.2.2	CUSUM Chart.....	366
19.2.3	EWMA Chart.....	367
19.2.4	Control Charts by Change-Point Detection (CPD).....	368
19.3	The Proposed SPC Chart	369
19.3.1	A Computationally Efficient Approach to Estimate $\tau^*(t)$	370
19.3.2	Integration of Ansari-Bradley Test with Cramer-von Mises Test.....	370
19.3.3	Data Pruning Based on P-Values.....	371
19.3.4	The Algorithm of the Proposed Control Chart.....	371
19.3.5	Implementation.....	371
19.4	Numerical Studies	372
19.4.1	Location Changes.....	372
19.4.2	Scale Changes.....	373
19.4.3	Location-Scale Changes.....	374
19.4.4	General Distributional Changes.....	374
19.5	Analysis of Various Real-World Data	376
19.5.1	Climate Data on Minneapolis, USA.....	377
19.5.2	Blood Glucose Monitoring Data.....	379
19.6	Concluding Remarks	379
	References	380

first briefly discuss a few commonly used SPC charts along with relevant references and then present a new chart for univariate continuous processes. Unlike most SPC charts in the literature, it neither assumes any “in-control” probability distribution nor requires any “in-control” Phase I data, and it aims to detect arbitrary distributional change. This chart uses a computationally efficient method to find the possible change-point. Moreover, the proposed chart uses a p-value-based data pruning approach to further increase the efficiency, and it combines the strengths of two different tests of hypotheses, which has a potentially broad application. Numerical simulations and two real-data analyses show that the chart can be used in various monitoring problems when the nature of distributional change is unknown.

Keywords

Ansari-Bradley test · Cramer-von Mises test · Distributional change · Nonparametric SPC · Pruning · P-value

Abstract

In the current era of computers, statistical monitoring of sequential observations is an important research area. In problems such as monitoring the quality of industrial products, health variables, climatological variables, etc., we are often interested in detecting a change in the process distribution in general, not just in mean or variance. We

19.1 Introduction

Statistical process control (SPC) charts are widely used for monitoring the stability of certain sequential processes in various disciplines including manufacturing and healthcare systems. Typically, the SPC charts assume that there are two causes of variability in the process measurements: one is “common cause” which is due to unavoidable randomness and another is “special cause” when an undesirable variability intervenes, for example, mechanical defects, improper handling of machines, human errors, onset of certain medical conditions, etc. When the variability is only due to common causes, the process is said to be “in-control”. “In-control” process measurements can be considered as realizations of

P. S. Mukherjee (✉)
Interdisciplinary Statistical Research Unit (ISRU), Indian Statistical Institute (ISI), Kolkata, West Bengal, India
e-mail: psm@isical.ac.in

a random model, for example, independent and identically distributed (i.i.d.) observations from a cumulative distribution function (c.d.f.) F_1 . When a special cause interferes, the process measurements no longer appear as i.i.d. realizations of F_1 , and then the system is said to be “out-of-control”. Practitioners typically divide SPC into two phases. Initially, a set of process measurements are analyzed in Phase I. If any “unusual” patterns in the process measurements are found, they make necessary adjustments and fine-tuning of the system. After removing all such special causes, we have a clean set of process measurements data under stable operating conditions, and they are representative of the actual process performance. The major goal of Phase II SPC control charts is to detect any change in process distribution after an unknown time-point.

A change in the process distribution may not always be in location and scale only; it can be general, for example, changes in degrees of freedom of chi-squared distribution. Furthermore, changes can either be *isolated*, i.e., the system goes “out-of-control” for a short time and then returns to “in-control”, or *persistent*, i.e., once the system goes “out-of-control,” it remains “out-of-control” or even goes further away from control until the special causes are removed. Among existing SPC charts in the literature, Shewhart-type [1] charts are used to detect *isolated* changes and cumulative sum (CUSUM) type charts (e.g., [2]) are used to detect *persistent* changes. However, most SPC charts consider shifts in location and/or scale, because they are most common and often capture other departures. In real-world problems, shifts in skewness and kurtosis can happen without much change in location and scale. For example, the shape of the process distribution gradually changing over time without much change in mean or variance. If we fail to detect those changes and let the process run, it can eventually become worse and a shift in location and scale can creep in. Moreover, the special causes that initiated the change can cause more damage to the system, and it may become a challenge to fix. If we can detect such a change in skewness, we can avoid subsequent troubles. Therefore, it is imperative to develop an SPC chart that can detect changes in the process distribution in general. The proposed chart precisely focuses on this.

Various types of SPC control charts have been proposed in the literature including Shewhart-type charts [1], cumulative sum (CUSUM)-type charts (e.g., [2]), exponentially weighted moving average (EWMA chart) [3] (e.g., [4, 5]), etc. Many control charts assume that “in-control” process distribution F_1 is either known or has a known parametric form (e.g., normal). In real-life problems, this is usually not the case. It has been demonstrated in the literature that the SPC charts using prespecified distribution in their designs may not be reliable in such cases (e.g., [6, 7]). To address this, a number of non-parametric or distribution-free SPC charts have been proposed. For example [8–18], and so on. [19] provides

an overview of nonparametric SPC charts. Discussions for multivariate cases are provided by Qiu and Hawkins [20, 21] and Qiu [22]. Some SPC charts (e.g., [23–25]) monitor process mean and variance jointly. Moustakides [26] proposes a method to detect distributional changes when both “in-control” and “out-of-control” distributions are known. Ross and Adams [27] propose two nonparametric control charts to detect arbitrary distributional change under the change point detection (CPD) framework when both “in-control” and “out-of-control” distributions are unknown. Mukherjee [28] also proposes a Phase II nonparametric SPC chart for detecting arbitrary distributional change, but it requires a substantial amount of “in-control” Phase I data. A thorough literature review on SPC charts can be found in [29] as well as in [30].

Most existing SPC charts mentioned above focus to detect changes in process location and scale but do not consider an arbitrary distributional change. Moreover, some methods require multiple observations at each time-point; some others require “in-control” Phase I observations. All of these may not be reasonable in many real-life problems. The proposed chart focuses on univariate continuous processes and proposes a p-value-based nonparametric SPC chart to detect an arbitrary distributional change when we can assume that the observations we collect are independent, but “in-control” c.d.f. F_1 is unknown, and “in-control” Phase I data are unavailable. p-value-based SPC charts are new trends as some researchers have already started developing those for better interpretation, e.g., [31, 32], etc.

Additionally, the proposed chart uses the strengths of Cramer-von Mises test and Ansari-Bradley test and demonstrates a better performance in detecting an arbitrary distributional change early. It is demonstrated in the literature [27] that the power of two-sample Cramer-von Mises test to detect a small change in standard deviation or scale is rather weak. The proposed chart overcomes that by integrating this test with Ansari-Bradley test, a nonparametric test for detecting scale change. The integration process is quite general in nature, and hence similar integration techniques can be used to design better control charts in many scenarios.

The major steps of the proposed SPC chart are the following: First, we estimate the possible change-point that gives the minimum p-value of the relevant two-sample Cramer-von Mises test statistic. We use a computationally efficient method to do this. Using that change-point, we calculate the p-value of Ansari-Bradley test and pick the smallest of the two p-values as the effective p-value and use it as the charting statistic. Since it is demonstrated in the literature (e.g., [27]) that Cramer-von Mises test does not have strong power to detect scale changes, if there is indeed a scale change in the process distribution, Ansari-Bradley test gives a smaller p-value than Cramer-von Mises test. Therefore, to detect such changes, it is better to use the smaller of the p-values. If

the process distribution changes without affecting the scale parameter, Cramer-von Mises test gives a smaller p-value than Ansari-Bradley test. It is also better to use the smaller of the two p-values to detect such a change as well. If the p-value is large enough, and our effective sequence of observations is too long, we prune observations from distant past and we collect the next observation. The amount of pruning is a nondecreasing function of the effective p-value as the higher the p-value, the more likely it is that no distributional change has occurred, and it is therefore better to ignore information from distant past to speed up computation. If the p-value is small enough, the chart signals a distributional change.

The remainder of the chapter is organized as follows. At the end of this paragraph, a nomenclature subsection is provided so that this chapter can be read smoothly. Next, brief descriptions of traditionally used SPC charts are provided in Sect. 19.2. The proposed control chart is described in Sect. 19.3. Numerical studies to evaluate its performance in comparison with several existing control charts are presented in Sect. 19.4. A climatological data analysis and a blood sugar monitoring data analysis by the proposed chart and its competitors are presented in Sect. 19.5. A few remarks in Sect. 19.6 conclude this chapter.

19.2 Traditionally Used SPC Charts

Statistical process control of a production process is roughly divided into two phases: Phase I and Phase II. In Phase I, i.e., in the initial stage, we usually do not have enough information about the performance of the production process, and our major goal in this stage is to adjust the production process so that it can run in a stable manner. First, we usually let the process produce a given amount of products, and then the quality characteristics of these products are analyzed. If the statistical analysis of these data shows indication that the process is not running stably, we try to figure out the root causes for that and make adjustments of the process so that it can run stably. After the adjustments, another set of data is collected and analyzed, and the production process should be adjusted again, if necessary. The analysis and adjustment process is iterated several times until we are confident that the performance of the production process is stable. Once all “special causes” have been removed and the production process is “in-control,” we collect an “in-control” dataset from the products manufactured under the stable operating conditions, and the “in-control” data are used for estimating the “in-control” distribution of the quality characteristic(s) of interest. Based on the actual “in-control” distribution if known, or estimated “in-control” distribution otherwise, a Phase II SPC control chart is designed. It is typically used for online monitoring of the production process. When the chart detects a significant change in the distribution of the quality

characteristic(s) from “in-control” distribution, it gives a signal and the production process is stopped immediately for identification of the root cause for such a change and its removal. This online monitoring stage is often called Phase II SPC.

In both Phase I and Phase II of SPC, many statistical tools such as histograms, stem-and-leaf plots, regression, and design of experiments are very helpful. Among all these, control charts are especially useful since they are constructed specifically to detect “out-of-control” performance of the production process. A charting statistic should be chosen such that it contains as much of the information in the observed data about the distribution of the quality characteristic(s) as possible and be sensitive to any distributional change as well. In the literature, different types of control charts have been developed including the Shewhart charts, the cumulative sum (CUSUM) charts, the exponentially weighted moving average (EWMA) charts, charts based on change-point detection (CPD), and so on. Brief descriptions of some of these control charts are provided below.

19.2.1 Shewhart Chart

The first control chart was proposed by Shewhart [1] in 1931. The chart assumes that the quality variable X follows normal distribution $N(\mu_0, \sigma^2)$, and at each time-point we obtain m independent quality observations. Denote $(X_{n1}, X_{n2}, \dots, X_{nm})$ to be the n -th batch of observations, where the batch size is $m \geq 2$. Traditional z-test is used to check if the process observations are “in-control” at the n -th time-point. The process is considered “out-of-control” when

$$\bar{X}_n > \mu_0 + z_{1-\alpha/2} \frac{\sigma}{\sqrt{m}} \quad \text{or} \quad \bar{X}_n < \mu_0 - z_{1-\alpha/2} \frac{\sigma}{\sqrt{m}},$$

where \bar{X}_n is the sample mean of $(X_{n1}, X_{n2}, \dots, X_{nm})$ and $z_{1-\alpha/2}$ is $(1 - \alpha/2)$ -th quantile of the standard normal distribution. This version can be used when both μ_0 and σ are known. However, it is usually not the case in reality. In that case, they have to be estimated from a dataset known to be “in-control.” Suppose, $(X_{i1}^*, X_{i2}^*, \dots, X_{im}^*), i = 1, 2, \dots, M$ be an “in-control” dataset. Let \bar{X}_i^* and R_i^* be the sample mean and sample range of the i -th batch of “in-control” dataset and \bar{X}^* and \bar{R}^* be the sample means of $\{\bar{X}_i^*, i = 1, 2, \dots, M\}$ and $\{R_i^*, i = 1, 2, \dots, M\}$, respectively. It can be easily verified that \bar{X}^* is an unbiased estimator of μ_0 and $\bar{R}^*/d_1(m)$ is an unbiased estimator of σ , where $d_1(m) = E(R_i^*/\sigma)$ is a constant depending on m . When $m = 2$, $d_1(m) = 1.128$, when $m = 5$, $d_1(m) = 2.326$. $d_1(m)$ values for many other commonly used m are provided in Table 3.1 of [30]. Therefore, the Shewhart chart signals a shift in process mean if

$$\begin{aligned} \bar{X}_n &> \bar{X}^* + z_{1-\alpha/2} \frac{\bar{R}^*}{d_1(m)\sqrt{m}} \quad \text{or} \\ \bar{X}_n &< \bar{X}^* - z_{1-\alpha/2} \frac{\bar{R}^*}{d_1(m)\sqrt{m}}. \end{aligned} \tag{19.1}$$

Traditionally, the manufacturing industry uses $\alpha = 0.0027$, and hence $(1 - \alpha/2)$ -th quantile of $N(0, 1)$, i.e., $z_{1-\alpha/2}$ is 3. Therefore, the chart signals mean shift at time-point n if \bar{X}_n falls outside the interval of width six sigma centered at μ_0 and where sigma is the standard deviation of \bar{X}_n . Thus, the terminology ‘‘six sigma’’ originated in the domain of quality control.

The performance of a control chart is traditionally measured by average run length (ARL). Since the charts use control limits for making decision on process performance, an ‘‘in-control’’ process sometimes give false signals of distributional shift. This phenomenon is analogous to having type I error in hypothesis testing. The number of samples or batches collected from the initial time-point of consideration to the occurrence of first false ‘‘out-of-control’’ signal when the process remains ‘‘in-control’’ is called ‘‘in-control’’ run length. The mean of such a run length is called ‘‘in-control’’ average run length denoted as ARL_0 . On the other hand, the number of samples or batches collected from the time-point when the shift actually occurs to the time-point of signal of shift is called ‘‘out-of-control’’ run length. Its mean is called ‘‘out-of-control’’ average run length denoted as ARL_1 . The ideal situation is that, for a control chart, ARL_0 value is large and ARL_1 value is small. However, similar to type I and type II error probabilities in hypothesis testing, it is difficult to achieve. Usually, when ARL_0 is large, ARL_1 is also relatively large and vice versa. In SPC literature, we usually fix the ARL_0 value at a given level and compare the performances of the control charts by comparing how small their ARL_1 values are. In the \bar{X} Shewhart chart as described above, the distribution of ‘‘in-control’’ run length is clearly geometric with parameter α . $\alpha = 0.0027$ makes $ARL_0 = 1/\alpha = 370.37$. ARL_1 can also be computed easily as a function of the shifted mean.

In the literature, there are many versions of Shewhart chart. One of them uses sample standard deviation to estimate σ rather than sample range. In this case, the chart gives a signal for mean shift when

$$\begin{aligned} \bar{X}_n &> \bar{X}^* + z_{1-\alpha/2} \frac{\bar{S}^*}{d_3(m)\sqrt{m}} \\ \text{or } \bar{X}_n &< \bar{X}^* - z_{1-\alpha/2} \frac{\bar{S}^*}{d_3(m)\sqrt{m}}, \end{aligned} \tag{19.2}$$

where \bar{S}^* is the sample mean of the batch-wise sample standard deviations $\{S_i^*, i = 1, 2, \dots, M\}$, i.e.,

$$S_i^* = \sqrt{\frac{1}{m-1} \sum_{j=1}^m (X_{ij} - \bar{X}_i^*)^2} \quad \text{for } 1, 2, \dots, M,$$

and $d_3(m) = E(S_i^*/\sigma)$ is a constant depending on the value of m . Under the same setup, Shewhart charts were constructed to monitor process variability. Defining $d_2(m) = \sqrt{\text{Var}\left(\frac{R_i^*}{\sigma}\right)}$, and using $d_1(m) = E(R_i^*/\sigma)$, we can estimate $\sigma_{R_i^*}$ by $\frac{d_2(m)}{d_1(m)}\bar{R}^*$. Therefore, this version of Shewhart chart gives a signal for a change in variability if

$$R_n > \bar{R}^* + z_{1-\alpha/2} \frac{d_2(m)}{d_1(m)}\bar{R}^* \quad \text{or} \quad R_n < \bar{R}^* - z_{1-\alpha/2} \frac{d_2(m)}{d_1(m)}\bar{R}^*. \tag{19.3}$$

Similarly, another version of Shewhart chart was constructed using sample standard deviation instead of sample range. Using the result $\sigma_{S_i^*} = \sigma\sqrt{1 - d_3^2(m)}$ proved by Kenney and Keeping [34], $\sigma_{S_i^*}$ can be estimated by $\frac{\bar{S}^*}{d_3(m)}\sqrt{1 - d_3^2(m)}$. Therefore, the chart gives a signal of a change in variability if S_n , the sample standard deviation of $(X_{n1}, X_{n2}, \dots, X_{nm})$, satisfies the following condition:

$$\begin{aligned} S_n &> \bar{S}^* + z_{1-\alpha/2} \frac{\sqrt{1 - d_3^2(m)}}{d_3(m)}\bar{S}^* \\ \text{or } S_n &< \bar{S}^* - z_{1-\alpha/2} \frac{\sqrt{1 - d_3^2(m)}}{d_3(m)}\bar{S}^*. \end{aligned} \tag{19.4}$$

Using the result that $\frac{(m-1)S_i^{*2}}{(\sigma^2)} \sim \chi_{m-1}^2$, another chart was constructed using sample variance instead of sample standard deviation. This chart gives a signal for a change in process variance when

$$S_n^2 > \bar{S}^{*2} \frac{\chi_{1-\alpha/2, m-1}^2}{m-1} \quad \text{or} \quad S_n^2 < \bar{S}^{*2} \frac{\chi_{\alpha/2, m-1}^2}{m-1}, \tag{19.5}$$

where \bar{S}^{*2} is the sample average of $\{S_i^{*2}, i = 1, 2, \dots, M\}$.

Next, we discuss \bar{X} Shewhart chart for monitoring individual observations rather than batched observations in Phase I SPC. The idea is to artificially create grouped data by grouping consecutive observations. First, we fix the size of each group $\tilde{m} > 1$. Then, the first \tilde{m} observations form the first group, the next \tilde{m} observations form the second group, and so on. Next, we can apply \bar{X} Shewhart chart (19.1) on the

grouped data. However, one problem here is that consecutive groups are \tilde{m} time-points apart. Hence, it is difficult to know the process behavior at each time-point. To overcome this limitation, most people adopt the idea of moving windows. We artificially create grouped data as follows: Group 1 ($X_1, X_2, \dots, X_{\tilde{m}}$), Group 2 ($X_2, X_3, \dots, X_{\tilde{m}+1}$), and so on until Group $(n - \tilde{m} + 1)$ ($X_{n-\tilde{m}+1}, X_{n-\tilde{m}+2}, \dots, X_n$). Denote $MR_1, MR_2, \dots, MR_{n-\tilde{m}+1}$ to be the sample ranges of the $(n - \tilde{m} + 1)$ groups of data and \overline{MR} to be their sample mean. From the definition of $d_1(\tilde{m})$, we can estimate σ by $\overline{MR}/d_1(\tilde{m})$. Therefore, the \bar{X} Shewhart chart for monitoring individual observations gives a signal of mean shift when

$$X_n > \bar{X} + z_{1-\alpha/2} \frac{\overline{MR}}{d_1(\tilde{m})} \quad \text{or} \quad X_n < \bar{X} - z_{1-\alpha/2} \frac{\overline{MR}}{d_1(\tilde{m})}. \tag{19.6}$$

Similarly, R Shewhart chart for monitoring individual observations gives a signal of a variability shift when

$$MR_i > \overline{MR} + z_{1-\alpha/2} \frac{d_2(\tilde{m})}{d_1(\tilde{m})} \overline{MR} \\ \text{or} \quad MR_i < \overline{MR} - z_{1-\alpha/2} \frac{d_2(\tilde{m})}{d_1(\tilde{m})} \overline{MR}. \tag{19.7}$$

Here, we should check at all time-points that belong to the i -th group, i.e., from the i -th to the $(i + \tilde{m} - 1)$ -th time-points. Other Shewhart charts for monitoring individual observations were constructed similarly.

In many applications, quality characteristics are categorical. Now, we describe some Shewhart charts for monitoring such characteristics. After certain products are randomly chosen for monitoring purposes, they are classified in conforming and nonconforming products based on requirements on the quality characteristics. Now, we monitor the proportion of nonconforming products over time. We assume that when a production process is “in-control,” the true proportion of nonconforming products is π_0 , and we obtain a random sample of m products at each time-point. Let Y be the number of nonconforming products obtained at a given time-point. Therefore, $Y \sim \text{Binomial}(m, \pi_0)$ when the process is “in-control.” Let $p = Y/m$ be the sample proportion of nonconforming products at the time-point. When m is large, the probability distribution of p can be well approximated by $N(\pi_0, \pi_0(1 - \pi_0)/m)$ by the central limit theorem when the process is “in-control.” Hence, the process can be called “out-of-control” if

$$p > \pi_0 + z_{1-\alpha/2} \sqrt{\frac{\pi_0(1 - \pi_0)}{m}} \\ \text{or} \quad p < \pi_0 - z_{1-\alpha/2} \sqrt{\frac{\pi_0(1 - \pi_0)}{m}}.$$

In practice, π_0 is often unknown and should be estimated from collected Phase I data, just like in the original version of \bar{X} Shewhart chart use estimated μ_0 and σ . As before, we assume that we have M batches of Phase I data. Let p_i^* be the sample proportion of nonconforming products in the i -th batch of Phase I data for $i = 1, 2, \dots, M$ and \bar{p}^* be their sample mean. Therefore, we can estimate π_0 by \bar{p}^* , and hence the p Shewhart chart gives a signal for a change in the proportion of nonconforming products at n -th time-point when

$$p_n > \bar{p}^* + z_{1-\alpha/2} \sqrt{\frac{\bar{p}^*(1 - \bar{p}^*)}{m}} \\ \text{or} \quad p_n < \bar{p}^* - z_{1-\alpha/2} \sqrt{\frac{\bar{p}^*(1 - \bar{p}^*)}{m}}. \tag{19.8}$$

There are other versions of Shewhart chart in the literature for monitoring count processes having distributions such as Poisson. Detailed descriptions of such charts are provided in Chapter 3 of [30].

Shewhart charts are good at detecting relatively large isolated shifts, but not so efficient in detecting relatively small but persistent shifts. This is because Shewhart charts evaluate the process performance based on the observed data collected at each individual time-point and ignore observations collected previously. Therefore, Shewhart charts are popular in Phase I SPC where large and isolated shifts are common but less commonly used in Phase II SPC.

As we mentioned before, the \bar{X} , R , and S Shewhart charts are appropriate to use only in cases when the process distribution is normal and the observations are independent of each other. When the process distribution is not normal, the probability of type I error, i.e., the probability of false “out-of-control” signal when the process is actually “in-control,” can substantially differ from the prefixed value of α . If the type I probability of a Shewhart chart is larger than α , then the chart will give false “out-of-control” signal more often than expected. Consequently, much time and resource will be wasted to find the cause of such signals and adjusting the related production process. On the other hand, if the type I probability of a Shewhart chart is smaller than α , then real shifts will be missed more often than expected and hence many nonconforming products could be manufactured. However, when the distribution of X_{ij} is non-normal but the batch size m is large, the issue described above will not be serious, because the distribution of \bar{X}_n can be well approximated by normal distribution due to the central limit theorem. In cases when the distribution of X_{ij} is non-normal and the batch size m is small, mainly two approaches are usually taken. One approach is to transform the non-normal data to normal and then use the conventional Shewhart charts to the transformed data [35,36], and the other approach is to use Shewhart charts that are constructed to monitor non-normal data.

If the performances of the Shewhart charts are evaluated by “in-control” average run length ARL_0 and “out-of-control” average run length ARL_1 , then the performance measures will be not accurate if the observations are correlated [37–39]. In these cases, correlations have to be handled properly.

19.2.2 CUSUM Chart

Shewhart chart makes a decision whether a process is “in-control” or not at a time-point using only the observations obtained at that time-point and ignoring all previous observations. Therefore, it is not very effective in Phase II monitoring in most cases, because previous observations contain helpful information about process performance at present. Page [2] suggested the first cumulative sum (CUSUM) chart to overcome this limitation. Let us first describe the basic CUSUM chart for detecting mean shift of a process following normal distribution. Again, the chart assumes that the process observations X_1, X_2, X_3, \dots follow $N(\mu_0, \sigma^2)$ and are independent. The CUSUM charting statistics are given by

$$C_n^+ = \max(0, C_{n-1}^+ ((X_n - \mu_0)/\sigma) - k), \quad (19.9)$$

$$C_n^- = \min(0, C_{n-1}^- ((X_n - \mu_0)/\sigma) + k) \quad (19.10)$$

where $C_0^+ = C_0^- = 0$, and $k > 0$ is an allowance constant. The CUSUM chart gives a signal of mean shift if

$$C_n^+ > \rho_c \quad \text{or} \quad C_n^- < -\rho_c \quad (19.11)$$

where $\rho_c > 0$ is a control limit. The allowance constant k is prespecified, and the value of ρ_c is chosen so that the average run length when the process is “in-control,” denoted as ARL_0 , equals a given number, say, 200, 370, 500, etc. Table 4.1 of [30] provides the values of ρ_c for various values of allowance constant k and ARL_0 . We can easily see that the charting statistics C_n^+ and C_n^- make use of all available data before the n -th time-point, and they restart from 0 when the cumulative charting statistics suggest no significant evidence for mean shift in the sense that $C_{n-1}^+ ((X_n - \mu_0)/\sigma) < k$ and $C_{n-1}^- ((X_n - \mu_0)/\sigma) > -k$. Because of this restarting mechanism, the CUSUM chart enjoys a good theoretical property that it has smallest “out-of-control” ARL, denoted as ARL_1 , among all control charts having the same value of ARL_0 . Moustakides [26] proved that the CUSUM chart with an allowance constant k has the shortest ARL_1 value among all charts with a fixed ARL_0 value for detecting a persistent shift of size $\delta = 2k$.

To be able to use the above CUSUM chart, the process observations should be independent before and after a potential shift, both “in-control” and “out-of-control” distributions have to be normal, and the parameters μ_0 and σ of the “in-control” distribution have to be known. However, in practice these assumptions may not be reasonable.

If the observations are autocorrelated, then the actual value of ARL_0 will be different from the specified value for which the control limit ρ_c is determined in case of i.i.d. (independent and identically distributed) normal observations as provided by Table 4.1 of [30]. For example, if the production process is autoregressive of order 1, and the autocorrelation is negative, then the actual value of ARL_0 will be larger than the specified value for which ρ_c is determined. Consequently, the chart will not be sensitive enough to mean shifts and a lot of nonconforming products can be manufactured. On the other hand, if the production process is autoregressive of order 1, with positive autocorrelation, then the actual value of ARL_0 will be smaller than the specified value for which ρ_c is determined. That means the chart will give too many false signals of mean shift than expected and the production process has to be stopped too many times unnecessarily, and hence many resources will be wasted. One commonly used approach to accommodate possible autocorrelation among observed data is to group neighboring observations into batches and then apply conventional CUSUM charts for independent data to the batch means. One major reason behind this idea is that possible autocorrelation in the original data will be mostly eliminated in the new batch means [40]. Because of autocorrelation, the standard deviation of standardized group means may not be 1, and it can differ very much from 1. Therefore, the actual ARL_0 may be far away from the specified ARL_0 value. To overcome this limitation of the grouping approach, the group means need to be scaled properly which is difficult to do unless we know the nature of correlation in the original data. For related descriptions, see [40, 41], and [42]. Another disadvantage of the grouping approach is that the control chart cannot detect a mean shift promptly as it has to wait until all observations within a group are obtained. An alternative approach to the grouping idea is to describe the correlation by a statistical model such as autoregressive moving average (ARMA) model. In many practical applications, appropriate special cases of the ARMA model, such as first-order autoregressive model, can be used; otherwise, an appropriate model can be selected by a model selection procedure. After a time-series model is chosen and fitted by a certain routine procedure in time-series analysis, we can calculate the residuals. If the chosen time-series model describes the observed “in-control” data adequately, and the production process is “in-control” until the given time-point, the residuals should approximately be independent with a zero-mean common normal distribution whose variance can be estimated by an appropriate estimator. Then, we can apply the conventional CUSUM chart to the calculated residuals. However, we should be careful that an “out-of-control” signal may not always be due to a shift in mean; it can be due to a change in correlation structure in the observations as well. Related discussions on model-based control charts for monitoring autocorrelated processes can be found in [43–45], and so on.

In cases when the observations are i.i.d. normal but the “in-control” mean μ_0 and variance σ^2 are unknown, a common approach is to estimate those from a large “in-control” dataset. However, even a small randomness of these estimated values affects the performance of the CUSUM charts in a significant way. Hawkins [46] explored this research question in detail. Since in many applications we cannot have an extremely large “in-control” dataset, Hawkins [46] proposed the self-starting CUSUM charts in which “in-control” parameters are estimated from the observations collected in Phase II SPC. Assume that no “out-of-control” signal is given until the $(n-1)$ -th time-point. A new observation is collected at the n -th time-point and we want to make a decision whether a signal of mean shift should be given at this time-point. Since no signal for mean shift is given at $(n-1)$ -th time-point, we can consider all observations collected at that time-point and before, i.e., $X_{n-1}, X_{n-2}, \dots, X_1$, as “in-control” observations. Therefore, μ_0 and σ^2 can be estimated by their sample mean \bar{X}_{n-1} and sample variance S_{n-1}^2 , respectively, as long as $n \geq 3$. Therefore, for constructing a CUSUM chart, it is natural to replace $(X_n - \mu_0)/\sigma$ in (19.9) and (19.10) by $T_n = (X_n - \bar{X}_{n-1})/S_{n-1}$. When X_1, X_2, \dots, X_n are i.i.d. $N(\mu_0, \sigma^2)$ which is the case when the process is “in-control” up to the n -th time-point, we can easily check that $(\sqrt{(n-1)/n})T_n \sim t_{n-2}$. As proved in [47], T_1, T_2, \dots, T_n are independent of each other when the process is “in-control” up to the n -th time-point. Therefore, in that case,

$$Z_n = \Phi^{-1} \left[\Upsilon_{n-2} \left(\sqrt{\frac{n-1}{n}} T_n \right) \right]$$

follow i.i.d. $N(0, 1)$ distribution, where Φ and Υ_{n-2} are cumulative distribution functions (c.d.f.) of $N(0, 1)$ and t_{n-2} , respectively. Since Φ^{-1} and Υ_{n-2} are increasing functions, a mean shift in the original observations X_i , $1 \leq i \leq n$ indicates a mean shift in the transformed observations Z_i , $1 \leq i \leq n$ and vice versa. Therefore, detection of mean shift in Phase II monitoring can be accomplished by using transformed observations Z_n , $n \geq 1$ in place of $(X_n - \mu_0)/\sigma$ in (19.9) and (19.10). It has been demonstrated in the literature that if a persistent mean shift occurs within the first few observations in Phase II monitoring, the self-starting CUSUM chart as described above has a weak power to detect it. Therefore, in practice, at least a dozen or more “in-control” observations should be collected in Phase II before using the self-starting CUSUM chart. Self-starting control charts are now popular in the literature ([48, 49], and so on).

The traditional CUSUM chart (19.9–19.10) has an allowance parameter k , which should be set as $\delta/2$ where δ is the size of potential mean shift. In practice, δ is often unknown at the time when we design the CUSUM chart and hence choice of k is not straightforward. Sparks [50] proposed two approaches to solve this issue. Sparks’ first approach is to use several CUSUM charts with different k

values simultaneously so that these charts target to detect mean shifts of different sizes. Such a joint control scheme gives an “out-of-control” signal of mean shift if at least one of the CUSUM charts detects a mean shift. Of course, the ARL_0 values of these CUSUM charts have to be the same prefixed number. If we have prior information about the potential mean shift, we should incorporate the information while determining the k values. Sparks’ second approach is to estimate the size of mean shift δ recursively at each time-point and updating the value of k accordingly. The control limit should also be updated at each time-point so that we can maintain the prespecified ARL_0 value. These are called adaptive CUSUM charts.

When the “in-control” distribution of the production process is not normal, then the traditional CUSUM charts should not be used. If we know that the “in-control” distribution is in exponential family such as gamma and Weibull distribution, then we can similarly construct CUSUM charts by using sequential probability ratio test. However, if the “in-control” distribution is completely unknown, we can use nonparametric control charts [11–18] that do not assume any “in-control” distribution.

The versions of CUSUM chart mentioned above were designed for detecting a step shift in process mean. However, in many applications, the process mean and/or variance changes gradually with or without a known parametric pattern, after the process becomes “out-of-control.” Such changes are called drifts. Gan [51], Davis and Woodall [52], and many other researchers proposed CUSUM charts for detecting linear drifts.

Recently, CUSUM charts with variable sampling rate become popular ([53, 54] and many others). In this type of CUSUM chart, the sampling rate varies over time based on all observed data. There are many different types of sampling rate such as variable sampling intervals, variable sample sizes, etc. One major advantage of variable sampling rate CUSUM charts compared to fixed sampling rate CUSUM charts is faster detection of small to moderate shift in process mean. Recently, Li and Qiu [32] suggested implementing a CUSUM chart using statistical p-values and proposed the concept of dynamic sampling.

In the literature, researchers have constructed CUSUM charts for monitoring the variance of the process distribution as well.

19.2.3 EWMA Chart

In spite of having good theoretical properties, CUSUM charts were difficult to use in the 1950s when there was no computers. A simpler chart, called exponentially weighted moving average (EWMA) chart, was proposed by Roberts [55] in 1959. Under the same assumptions and notations of CUSUM chart, the EWMA charting statistic is defined as

$$E_n = \lambda X_n + (1 - \lambda)E_{n-1} \quad (19.12)$$

where $E_0 = \mu_0$ and $0 < \lambda \leq 1$ is a weighting parameter. We can easily check that

$$E_n = \lambda \sum_{i=1}^n (1 - \lambda)^{n-i} X_i + (1 - \lambda)^n \mu_0, \quad (19.13)$$

and when the process is “in-control,”

$$E_n \sim N\left(\mu_0, \frac{\lambda}{2 - \lambda} [1 - (1 - \lambda)^{2n}] \sigma^2\right).$$

That means E_n is a weighted average of μ_0 and all observations up to time-point n , and the weight received by X_i decays exponentially fast when i moves away from n . Therefore, it becomes easy to study the properties of E_n when the process is “in-control.” From the probability distribution of E_n , EWMA chart gives a signal for mean shift when

$$|E_n - \mu_0| > \rho_e \sigma \sqrt{\frac{\lambda}{2 - \lambda} [1 - (1 - \lambda)^{2n}]}, \quad (19.14)$$

where $\rho_e > 0$ is a control limit. $\lambda > 0$ is chosen beforehand, and the value of ρ_e is determined such that a specified value of ARL_0 is achieved.

To be able to use the EWMA chart in practice, we need to choose λ values properly and then the value of ρ_e so that the prespecified ARL_0 value is achieved. Just like the CUSUM charts, we need to specify a target shift size first and then we can search for a λ value and the corresponding ρ_e value such that the prespecified ARL_0 value is achieved and the ARL_1 value for detecting the mean shift of target size is minimized. As a general guideline, small λ values are good for detecting relatively small mean shifts, and large λ values are good for detecting relatively large means shifts. Crowder [56] provides a discussion on this issue. While we assume that the “in-control” process distribution is normal, some researchers have demonstrated that the EWMA chart is quite robust to normality assumption [3].

In case of autocorrelated observations, similar approaches as we described in the CUSUM chart can be implemented in case of EWMA charts as well. However, some researchers suggest applying the EWMA charts directly to the original data and then adjusting the control limits to reflect the impact of autocorrelations [57].

For using Shewhart, CUSUM, or EWMA chart as described before, μ_0 and σ has to be known or estimated before the monitoring starts. As we have discussed before, this is not convenient in many applications, and hence we should use self-starting control charts. Just like the self-starting CUSUM control charts, we first transform the original data to Z_i , $i \geq 1$ and then apply the traditional EWMA chart to the

transformed data. In the literature, there are several methods such as [58] where λ is chosen adaptively depending on the size of potential mean shift. These are called adaptive EWMA charts.

In the literature, EWMA charts have been developed when the process distribution follows other parametric forms. Some researchers [59, 60] constructed EWMA charts for monitoring processes following Weibull distributions. Borrer et al. [61], Gan [62], and some others discussed process monitoring when the process distribution is Poisson. Perry and Pignatiello [63], Sparks et al. [64], and many others discussed EWMA process monitoring with binomial or negative binomial distributions.

All versions EWMA charts that we discussed so far are designed for detecting step shift in the process mean. In some practical situations, when the process becomes “out-of-control,” its mean departs gradually from the “in-control” level. It is important that we can detect such gradual departures, called drifts, as early as possible. In the literature, some researchers have modified EWMA charts to detect drifts efficiently ([65] and others).

Just like CUSUM charts, researchers have constructed EWMA charts to monitor the variance of the production processes as well.

19.2.4 Control Charts by Change-Point Detection (CPD)

In change-point detection (CPD), the distribution of the first part of a sequence of random variables is assumed to be the same, the distribution of the remaining part of the random variables of that sequence is also assumed to be the same, but the distributions of the two parts are assumed to be different. The specific position in the sequence at which the distribution of the random variables changes from one to the next is called a change-point. Our major goal is to estimate the position of the change-point. Gombay [66], Hinkley [67], and many others in the literature provide detailed description of this topic. In change-point detection problems, the sample size is usually fixed. In Phase I SPC, the sample size is usually fixed, and then the change-point methods can be applied directly. In Phase II SPC, observations are obtained sequentially over time. Therefore, change-point methods must be applied appropriately in such cases. Recently, change-point methods have been modified and applied to the SPC problems ([4, 5], and others) as well. Change-point-based control charts are good at detecting small and persistent shifts and can estimate the position of change-point efficiently.

Let us describe the change-point based control chart proposed by Hawkins et al. [4]. It assumes that the observations X_1, X_2, \dots, X_n follow this change-point model

$$\begin{aligned}
 X_i &= \mu_0 + \varepsilon_i & \text{for } 1 \leq i \leq r \\
 X_i &= \mu_1 + \varepsilon_i & \text{for } (r + 1) \leq i \leq n
 \end{aligned}$$

where r is the change-point and ε_i , $1 \leq i \leq n$ is a sequence of i.i.d. random variables having the common distribution $N(0, \sigma^2)$. For testing the existence of the change-point, the likelihood ration test statistic is

$$T_{max,n} = \max_{1 \leq j \leq n-1} \sqrt{\frac{j(n-j)}{n}} \left| \bar{X}_j - \bar{X}'_j \right| / \tilde{S}_j \quad (19.15)$$

where \bar{X}_j and \bar{X}'_j are sample means of first j and the remaining $(n-j)$ observations, and $\tilde{S}_j^2 = \sum_{i=1}^j (X_i - \bar{X}_j)^2 + \sum_{i=j+1}^{n-j} (X_i - \bar{X}'_j)^2$. The change-point-based chart gives an “out-of-control” signal of mean shift when

$$T_{max,n} > \rho_n \quad (19.16)$$

where $\rho_n > 0$ is a control limit. If we have an “out-of-control” signal, then the position of change-point r is estimated by the maximizer in (19.15). Hawkins et al. [4] provided formulas to calculate the values of ρ_n for commonly used prespecified values of ARL_0 .

Just like other control charts, change-point-based charts were also constructed to accommodate other situations such as if the “in-control” process distribution is not normal, but other known parametric distribution. Change-point-based charts were also developed to monitor process variance as well.

Although the change-point-based control charts have their advantages over Shewhart, CUSUM, and EWMA charts, their computation is still relatively complex. This is because the estimator of the position of change-point has to be recalculated each time a new observation is obtained which involve a search in a sequence of n time-points. A systematic comparison between the performances of self-starting and adaptive traditional (CUSUM and EWMA) control charts and the change-point-based control charts is currently lacking in the literature.

In practice, many variables do not follow normal distribution. For example, many economic indices, rainfall distribution, lifetime distribution of many products, etc. are usually skewed to the right. In most instances it is difficult to find an appropriate parametric distribution for their modeling. Researchers use nonparametric methods to describe their distributions. When the normality assumption is not reasonable, several researchers have pointed out that the traditional control charts would be unreliable for process monitoring ([6,7,16], and many others). In such cases, nonparametric statistical methods based on the ranking or ordering information of the observations can be considered for making inferences about the underlying process distribution. Another approach is based on data categorization. Clearly, both approaches

have the limitation of losing useful information during ranking and categorization. However, the methods based on data categorization seem to be more efficient in process monitoring as use some information about observation magnitudes. Nonparametric versions of conventional charts such as Shewhart, CUSUM, EWMA, and change-point-based have been developed in the literature. Most nonparametric control charts in the literature are for Phase II SPC. However, there is only a limited discussion on Phase I SPC when the process distribution does not follow any common parametric form [68,69].

Many researchers have developed control charts in the literature to jointly monitor process mean and variance as well. We skip such details in this chapter. We also omit descriptions of multivariate control charts as well. Interested readers should go through [30].

19.3 The Proposed SPC Chart

In this section, we describe our proposed Phase II chart for detecting persistent distributional change in univariate continuous processes. We assume that “in-control” probability distribution of the process is unknown, “in-control” Phase I data are unavailable, and the observations we collect are independent of each other. Let $Y_1, Y_2, \dots, Y_t, \dots$ be a sequence of independent observations during Phase II process monitoring. We start performing statistical monitoring right after time-point $S_0 \geq 4$, because we need to need have at least a few observations so that statistical tests have enough power. This is a common practice as discussed in [4,27], etc. At each time $t \geq S_0$, we consider the following change-point framework. Assuming τ , $2 \leq \tau \leq t - 2$ to be a possible point of distributional change, we consider two samples $\{Y_1, Y_2, \dots, Y_\tau\}$ and $\{Y_{\tau+1}, Y_{\tau+2}, \dots, Y_t\}$, and perform Cramer-von Mises test to check if they are coming from the same unknown continuous cumulative distribution function (cdf). For statistically testing whether two samples are from the same continuous distribution, two tests are commonly used: Kolmogorov-Smirnov test and Cramer-von Mises test. Kolmogorov-Smirnov test does not work well if there are ties in the samples. Also, Cramer-von Mises test usually have more power than Kolmogorov-Smirnov test in many situations. Therefore, we use Cramer-von Mises test ahead of Kolmogorov-Smirnov test. The two-sample Cramer-von Mises test statistic is given by

$$\begin{aligned}
 C_\tau(t) &= \frac{\tau(t-\tau)}{t^2} \left(\sum_{i=1}^{\tau} (F^*(Y_i) - G^*(Y_i))^2 \right. \\
 &\quad \left. + \sum_{j=\tau+1}^t (F^*(Y_j) - G^*(Y_j))^2 \right), \quad (19.17)
 \end{aligned}$$

where F^* and G^* are the empirical distributions associated with the samples $\{Y_1, Y_2, \dots, Y_\tau\}$ and $\{Y_{\tau+1}, Y_{\tau+2}, \dots, Y_t\}$. Large values of C_τ means $\{Y_1, Y_2, \dots, Y_\tau\}$ and $\{Y_{\tau+1}, Y_{\tau+2}, \dots, Y_t\}$ are possibly coming from different cdfs. Suppose $P_\tau^{(CvM)}(t)$ is the p-value of the two-sample Cramer-von Mises test. The details of computing $P_\tau^{(CvM)}(t)$ are provided in Sect. 19.3.5 along with other computational technicalities of the proposed chart. A *natural approach* in this framework is to (i) calculate the p-values of two-sample Cramer-von Mises test for all possible values of τ , i.e., $\tau = 2, 3, \dots, t-2$, (ii) determine $\tau^*(t) = \arg \min_{\tau \in \{2, 3, \dots, (t-2)\}} P_\tau^{(CvM)}(t)$, and (iii) use $P_{\tau^*}^{(CvM)}(t)$ as the charting statistic.

However, there are *two major drawbacks* of this approach. *The first drawback* is that the determination of $\tau^*(t)$ can be computationally expensive because we have to compute $P_\tau^{(CvM)}(t)$ for all possible values of τ , i.e., $2, 3, \dots, (t-2)$. Moreover, we need to execute similar procedures for all values of $t \geq S_0$ until we get a signal. We run into this issue in the change-point framework wherever we need to compute the relevant statistic for each value of the possible change-point. To reduce computation, we consider the following two techniques and both can be applied to the proposed SPC chart. The first technique is to estimate $\tau^*(t)$ efficiently by the method proposed in Sect. 19.2.1 which can be applied to other change-point problems of similar nature. The second technique is the pruning of data from distant past based on p-values as proposed by Mukherjee [28]. *The second drawback* is the weakness of the power of Cramer-von Mises test to detect a change in standard deviation or scale. The proposed chart addresses this issue by integrating Cramer-von Mises test with Ansari-Bradley test which is a nonparametric test to detect scale difference. We describe these approaches below.

19.3.1 A Computationally Efficient Approach to Estimate $\tau^*(t)$

We note that the values of $P_\tau^{(CvM)}(t)$ and $P_{\tau'}^{(CvM)}(t)$ should not be too different as long as τ and τ' are close. In other words, for each t , the values of $P_\tau^{(CvM)}(t)$ for different τ are strongly autocorrelated. Therefore, we can make the procedure faster by first calculating $P_\tau^{(CvM)}(t)$ for τ -values that are multiples of $\lfloor \sqrt{t} \rfloor$ instead of all possible values of τ . For a demonstration purpose, consider the case when $t = 100$. Calculate $P_\tau^{(CvM)}(t)$ for $\tau = 10, 20, \dots, 90$ instead of $\tau = 2, 3, 4, \dots, 98$. Then, select the value of τ for which $P_\tau^{(CvM)}(t)$ is minimum, and calculate $P_\tau^{(CvM)}(t)$ for τ -values in the two adjoining intervals. In other words, if $\tau = 30$ gives the smallest $P_\tau^{(CvM)}(t)$ -value among $\tau = 10, 20, 30, \dots, 90$, compute $P_\tau^{(CvM)}(t)$ for $\tau = 21, 22, \dots, 29, 30, 31, \dots, 39$ and select the τ -value for which $P_\tau^{(CvM)}(t)$ is the smallest. The detailed procedure to estimate $\tau^*(t)$ is described below.

For each t , instead of calculating $P_\tau^{(CvM)}(t)$ for possible values of τ , i.e., $2, 3, \dots, (t-2)$, calculate the statistic for $\tau = i \cdot \lfloor \sqrt{t} \rfloor$, where possible values of i are $1, 2, \dots, I(t)$, where $I(t)$ is the largest integer for which $I(t) \cdot \lfloor \sqrt{t} \rfloor \leq (t-2)$. Here, $\lfloor \sqrt{t} \rfloor$ is the largest integer smaller than or equal to t . Since we start monitoring when the time-point $t \geq S_0 \geq 4$, we always have at least one positive integer i for which we can compute $P_\tau^{(CvM)}(t)$. For example, when $t = S_0 = 4$, we have to compute $P_\tau^{(CvM)}(t)$ for only one value of τ , i.e., when $\tau = 2$. Next, we find $\tilde{\tau} = \arg \min_{i \in \{1, 2, \dots, I(t)\}} P_{i \cdot \lfloor \sqrt{t} \rfloor}^{(CvM)}$. Since the $\tau^*(t)$ should be close to $\tilde{\tau}$, we calculate $P_\tau^{(CvM)}(t)$ for all integer values of τ within $[\max\{\tilde{\tau} - \lfloor \sqrt{t} \rfloor + 1, 2\}, \min\{\tilde{\tau} + \lfloor \sqrt{t} \rfloor - 1, (t-2)\}]$ and pick the integer within that interval for which $P_\tau^{(CvM)}(t)$ is minimum. This is our estimated $\tau^*(t)$ and we call it $\tilde{\tau}^*(t)$. The method is summarized as follows:

- (i) Calculate $P_\tau^{(CvM)}(t)$ for $\tau = i \cdot \lfloor \sqrt{t} \rfloor$, where possible values of i are $1, 2, \dots, I(t)$, where $I(t)$ is the largest integer for which $I(t) \cdot \lfloor \sqrt{t} \rfloor \leq (t-2)$.
- (ii) Find $\tilde{\tau} = \arg \min_{i \in \{1, 2, \dots, I(t)\}} P_{i \cdot \lfloor \sqrt{t} \rfloor}^{(CvM)}$.
- (iii) Calculate $P_\tau^{(CvM)}(t)$ for all integer values of τ within the interval:

$$[\max\{\tilde{\tau} - \lfloor \sqrt{t} \rfloor + 1, 2\}, \min\{\tilde{\tau} + \lfloor \sqrt{t} \rfloor - 1, (t-2)\}].$$

- (iv) Estimate $\tau^*(t)$ by the integer within the interval in (iii) for which $P_\tau^{(CvM)}(t)$ is minimum.

19.3.2 Integration of Ansari-Bradley Test with Cramer-von Mises Test

It is well documented in the literature (e.g., [27]) that Cramer-von Mises test does not have high power to detect changes in scale parameters. However, it has high power to detect changes in location parameters. Because of this weakness, we integrate Cramer-von Mises test with Ansari-Bradley test [70] in the proposed control chart. Ansari-Bradley test is a nonparametric test based on rank sum to detect differences in scale parameters. The integration procedure is as follows. For each t , once $\tau^*(t)$ is estimated by the procedure in Sect. 19.2.1:

- (i) Record the p-value of two-sample Cramer-von Mises test for checking if $\{Y_1, Y_2, \dots, Y_{\tilde{\tau}^*}\}$ and $\{Y_{\tilde{\tau}^*+1}, Y_{\tilde{\tau}^*+2}, \dots, Y_t\}$ are realizations from the same continuous cdf. Obviously, it is $P_{\tilde{\tau}^*}^{(CvM)}(t)$.
- (ii) Perform Ansari-Bradley two-sample test for checking if the scale parameters of $\{Y_1, Y_2, \dots, Y_{\tilde{\tau}^*}\}$ and $\{Y_{\tilde{\tau}^*+1}, Y_{\tilde{\tau}^*+2}, \dots, Y_t\}$ are same. Call the p-value $P_{\tilde{\tau}^*(t)}^{(AB)}$.

(iii) Calculate $p^{(E)}(t) = \min\{P_{\hat{\tau}^*(t)}^{(\text{CvM})}(t), P_{\hat{\tau}^*(t)}^{(\text{AB})}(t)\}$, and use this as charting statistic.

When we integrate two different tests in a change-point-based chart, it is natural to perform change-point analysis for both tests, because the estimated change-points can be different for different tests. However, to avoid extra computation, we do not consider this approach. Numerical studies show that the proposed chart based on the procedures as described performs well in detecting changes in scale parameters. When there is a small change in scale parameters and the p-value of Cramer-von Mises test is not small enough to give a signal, it is still small and hence the change-points based on both tests should be close. The proposed chart just detects such change by integrating a more powerful test appropriately.

19.3.3 Data Pruning Based on P-Values

In case the proposed chart does not detect any distributional change for a long time, the sequence of “in-control” observations will be long, and hence calculation of $\hat{\tau}^*(t)$ as described in Sect. 19.2.1 can be time-consuming as well. If the influx of observations is rapid, then this chart in its present form cannot be used to monitor such processes. To solve this problem, consider datapruning from distant past. If $p^{(E)}(t)$, calculated by the procedure described in Sect. 19.2.2 is large, it is very unlikely that a distributional change has taken place. Therefore, we prune a few observations from distant past and focus on more recent observations. In this way, we make sure that the sequence of “in-control” observations does not become too long. We provide the description of the method below.

When $\tilde{t} \geq T_0$, where \tilde{t} is the current length of the sequence of observations, and T_0 is a threshold parameter of the SPC chart, we consider the possibility of data pruning. If $p^{(E)}(t) \geq P_0$, we prune the oldest $C(\tilde{t}, p^{(E)}(t), P_0) = \left\lceil \tilde{t} \cdot \min\left(0.2, \left(\frac{p^{(E)}(t) - P_0}{1 - P_0}\right)\right) \right\rceil$ number of observations. Here, we make sure not to prune too much in one step, by specifying that we cannot prune more than 20% of the current length of the sequence at one step. Here, we set the maximum amount of pruning at one step to be 20% based on the performance of the chart in our simulation experiments in terms of how fast it can compute and how early it can give a signal when a process goes “out-of-control.” While one can introduce a parameter for the maximum percentage we can prune at one step and select its value based on a reasonable criterion, we set it equal to 20% for simplicity. Once we prune, we can estimate $\tau^*(t)$ faster in the next step, i.e., after the arrival of next observation. The summary of data pruning is as follows:

- (i) When $p^{(E)}(t) < P_0$, give a signal for distributional change and stop process monitoring; otherwise, go to Step (ii).
- (ii) If $\tilde{t} \geq T_0$, prune the oldest $C(\tilde{t}, p^{(E)}(t), P_0) = \left\lceil \tilde{t} \cdot \min\left(0.2, \left(\frac{p^{(E)}(t) - P_0}{1 - P_0}\right)\right) \right\rceil$ number of observations, and collect the next observation. Otherwise, go to Step (iii).
- (iii) Do not prune and collect the next observation.

19.3.4 The Algorithm of the Proposed Control Chart

The summary of the procedures to run the proposed control chart is as follows:

1. When $\tilde{t} < S_0$, collect the next observation. Otherwise, go to Step 2.
2. Calculate $\hat{\tau}^*(t)$, an estimate of $\tau^*(t)$ by the method described in Sect. 19.2.1. Go to Step 3.
3. Calculate $P_{\hat{\tau}^*(t)}^{(\text{CvM})}(t)$, $P_{\hat{\tau}^*(t)}^{(\text{AB})}(t)$, and $p^{(E)}(t)$ by the method described in Sect. 19.2.2. If $p^{(E)}(t) < P_0$, give a signal for distributional change and stop process monitoring; otherwise, go to Step 4.
4. Perform the data pruning procedure as described in Sect. 19.2.3. Go to Step 1.

19.3.5 Implementation

Calculation of p-values for two-sample Cramer-von Mises tests is computationally expensive and time-consuming especially when the sample sizes are large [71]. However, [72] provide the asymptotic distribution of two-sample Cramer-von Mises criterion. We also note that the convergence rate is rapid. Therefore, in our implementation, we use asymptotic p-values rather than the exact ones. Using the software R (<https://www.r-project.org/>) and the R-package *CvM2SL2Test* developed by Xiao [73], we extend the table provided by Anderson and Darling [72] to calculate p-values of two-sample Cramer-von Mises tests when they are smaller than 0.01. We use the same R-package *CvM2SL2Test* to calculate the two-sample Cramer-von Mises test statistics given as in (1) and then calculate their asymptotic p-values. For calculating (1), the R-package uses a C++ program developed by Xiao et al. [74]. We use R-function *ansari.test()* to perform Ansari-Bradley tests. It should be noted that approximate p-values are calculated even for small samples in presence of ties. The proposed method is designed to monitor univariate continuous

processes and hence cannot handle even a moderate number of ties in the observation sequence. However, this chart can still perform well in presence of a small number of ties.

19.4 Numerical Studies

We perform several numerical studies to evaluate the performance of the proposed method in comparison with a number of state-of-the-art change-point-based control charts. Our main goal is to find a chart that shows overall better performance to detect arbitrary distributional changes. We also study how the performances are affected by the number of “in-control” observations before the distributional change. Like most research articles in the literature, we also evaluate the performances of the control charts by the average number of observations, called “run length” after the first observation from the changed distribution is obtained. We call it ARL_1 . The smaller the value of ARL_1 , the better the performance. Of course, we set the control limits of all control charts such that the average number of “in-control” observations required to give a false signal when no distributional change takes place is a prefixed large number. We call it ARL_0 . In all our numerical simulations, we set $ARL_0 = 500$ unless mentioned otherwise.

We compare the proposed method with four other change-point-based methods. The first method is proposed by Hawkins and Deng [13]. It is based on Mann-Whitney test and it aims to detect location shift. We call this method MW. The next competing control chart, proposed by Ross et al. [33], is based on Mood test [75]. This chart aims to detect scale change in the process distribution. We call it Mood. The third competing chart, also proposed by Ross et al. [33], is based on Lepage test [76]. It aims to detect changes in both location and scale of the process distribution, and we call it Lepage. Our fourth competing chart, proposed by Ross and Adams [27], aims to detect arbitrary distributional change using Cramer-von Mises test. We call this chart CvM.

Initially, we consider three “in-control” process distributions: standard normal, $N(0, 1)$; standardized t-distribution with 2.5 degrees of freedom, $ST(2.5)$; and standardized log-normal distribution with parameters 1.0 and 0.5, $SLN(1.0, 0.5)$. Note that the mean and standard deviation of log-normal distribution with parameters 1.0 and 0.5 are 3.08 and 1.64, respectively. However, we approximate those by 3 and 1.6 while standardizing. We first consider shifts in location only, scale only, and both location and scale simultaneously. Finally, we consider arbitrary changes of various distributions. To compare the performances of various methods, we consider two cases: when the distribution change occurs early, right after

time-point $\tau = 50$, and when the change occurs late, right after time-point $\tau = 300$. It should be noted that if a false signal occurs before the actual distributional change, we disregard that sequence in our simulation, as it is a reasonable practice.

We use the R-package *cpm* to run the competing charts. For the proposed method, we need to select S_0 and T_0 . For all methods, we select the startup time to be 20, and for the proposed method, we select $S_0 = 20$ and $T_0 = ARL_0 = 500$. For comparing the performances of the methods when the distributional change is arbitrary, we run four versions of the proposed chart called P_{250} , P_{500} , P_{1000} , and P_∞ when $T_0 = 250, 500, 1000$, and ∞ , respectively. From Table 19.6, we see that $T_0 = 500$ is a reasonable choice considering the fact that larger values of T_0 requires more time for computation. It should be noted that the proposed chart that is designed to detect an arbitrary distributional change may not outperform the charts designed to detect a specific type of distributional changes when the actual distributional change is of that particular type. For example, MW should perform better than the proposed chart when the distributional change involves location change only. However, our goal is to find a chart that performs well to detect all types of distributional changes so that it can be used in various applications where the natures of changes are unknown.

19.4.1 Location Changes

First, we focus on changes in location only. We consider four different amounts of shift, $\delta_L = 0.25, 0.5, 1.0$, and 2.0 . For each of three distributions $N(0, 1)$, $ST(2.5)$, and $SLN(1.0, 0.5)$, we generate 50,000 sequences of observations where the post-change observations right after time-point τ are calculated by adding δ_L . From Table 19.1, we observe that

- MW and CvM are slightly better than the proposed method when δ_L is small or moderate. When δ_L is large, all methods except Mood perform well. The reason is that MW is designed to detect location changes only; therefore, it has better power when detecting location changes compared to the methods designed to detect arbitrary distributional changes. Mood is designed to detect scale changes; therefore, it cannot perform well in this case.
- Detections of location changes are faster when the “in-control” distribution is $ST(2.5)$ or $SLN(1.0, 0.5)$ compared to the $N(0, 1)$ case.
- Changes occurring right after time-point $\tau = 300$ are easier to detect compared to the changes occurring right after $\tau = 50$.

Table 19.1 Mean delay in detection of various location shifts δ_L occurring right after time τ . “In-control” distributions considered are $N(0, 1)$, $ST(2.5)$, and $SLN(1.0, 0.5)$. The results are based on 50,000 random simulations when $ARL_0 = 500$

	$\tau = 50$				$\tau = 300$			
	$\delta_L =$				$\delta_L =$			
$N(\delta_L, 1)$	0.25	0.50	1.00	2.00	0.25	0.50	1.00	2.00
MW	377.1	134.3	13.8	4.9	150.8	35.1	10.2	3.9
Mood	502.6	506.3	310.5	32.9	479.3	399.0	71.2	4.3
Lepage	434.4	225.6	20.0	3.7	228.6	49.5	11.5	3.1
CvM	387.7	153.9	14.8	4.3	168.3	37.8	10.7	3.9
Proposed	400.0	169.6	16.0	4.8	190.5	41.5	12.3	5.3
$ST(2.5) + \delta_L$								
MW	199.8	21.4	6.0	3.5	46.8	13.3	5.2	3.1
Mood	492.1	346.5	72.1	11.5	446.0	180.0	7.9	2.5
Lepage	299.2	38.3	6.4	2.4	74.9	16.9	4.5	2.1
CvM	195.5	20.6	5.7	3.2	46.3	13.1	4.9	3.0
Proposed	212.1	22.3	6.2	3.6	50.3	14.6	6.4	4.2
$SLN(1.0, 0.5) + \delta_L$								
MW	368.4	96.0	10.9	4.3	113.8	26.4	8.4	3.8
Mood	527.3	272.4	71.5	45.9	406.2	171.3	139.3	5.6
Lepage	411.3	93.9	15.9	4.4	143.3	37.6	11.9	3.3
CvM	378.1	103.3	10.7	4.0	122.3	27.0	8.3	3.6
Proposed	356.9	93.0	11.2	4.4	120.8	28.2	9.6	5.0

Table 19.2 Mean delay in detection of various scale changes δ_S occurring right after time τ . “In-control” distributions considered are $N(0, 1)$, $ST(2.5)$, and $SLN(1.0, 0.5)$. The results are based on 50,000 random simulations when $ARL_0 = 500$

	$\tau = 50$				$\tau = 300$			
	$\delta_S =$				$\delta_S =$			
$N(0, \delta_S^2)$	1.50	2.00	0.50	0.25	1.50	2.00	0.50	0.25
MW	320.2	226.4	753.0	843.5	154.0	73.7	1470.1	1929.3
Mood	94.6	17.1	37.2	11.2	26.9	10.6	21.3	10.1
Lepage	144.6	25.8	63.3	15.4	34.1	13.2	31.0	14.8
CvM	320.5	196.6	563.9	69.4	131.6	47.2	102.8	30.4
Proposed	195.2	37.8	84.6	13.2	59.1	24.4	31.9	13.8
$ST(2.5) \cdot \delta_S$								
MW	359.4	273.6	704.5	816.5	205.1	107.8	1272.2	1798.7
Mood	205.0	49.4	79.8	14.0	49.2	17.3	31.0	12.3
Lepage	246.8	85.3	141.3	19.7	64.0	21.3	45.6	17.9
CvM	348.9	252.5	613.3	140.6	182.4	68.6	161.4	38.7
Proposed	275.4	80.1	144.9	17.2	89.4	33.0	42.5	16.1
$SLN(1.0, 0.5) \cdot \delta_S$								
MW	274.5	167.1	671.4	660.7	102.7	47.1	721.6	485.9
Mood	66.2	13.8	28.8	12.0	20.4	8.8	20.1	11.3
Lepage	95.9	17.3	42.6	15.3	25.0	10.7	27.3	15.3
CvM	268.0	123.5	421.5	47.3	83.9	33.8	65.4	24.2
Proposed	142.1	27.3	55.7	12.6	45.1	20.3	26.7	13.5

19.4.2 Scale Changes

Now, we focus on changes in scale only. We consider four scale changes, namely, $\delta_S = 1.50, 0.50, 2.00,$ and 0.25 . For each of three distributions, $N(0, 1)$, $ST(2.5)$, and $SLN(1.0, 0.5)$, we generate 50,000 sequences of observations where the post-change observations are calculated by multiplying δ_S . From Table 19.2, we observe that

- Mood and Lepage are slightly better than the proposed method in some cases. The reason is that Mood is designed

to detect scale changes only and Lepage is designed to detect both location and scale change. Therefore, they have better powers when detecting scale changes compared to the methods designed to detect arbitrary distributional changes. MW is designed to detect location changes; therefore, it cannot perform well in this case. The proposed method performs much better than CvM. The reason is the incorporation of Ansari-Bradley test with Cramer-von Mises test.

- MW performs fairly well when $\delta_S > 1.00$ but performs very poorly when $\delta_S < 1.00$. One explanation of this is

when $\delta_S > 1.00$, extremely large or small numbers are more likely, and MW interprets those as location changes. This phenomenon is commented in [27] and in [24].

- Detections of scale changes are faster when the “in-control” distribution is $ST(2.5)$ and $SLN(1.0, 0.5)$ compared to the $N(0, 1)$ case.
- In this case also, changes occurring right after time-point $\tau = 300$ are easier to detect compared to the changes occurring right after $\tau = 50$. MW, however, cannot detect a decrease in scale at all.

- Lepage performs the best, as it should because it is designed to detect changes in location and scale simultaneously. Mood works well in some cases. The proposed method works well, better than CvM in many cases.
- In these case also, detections of changes in location and scale simultaneously are faster when the “in-control” distribution is $ST(2.5)$ and $SLN(1.0, 0.5)$ compared to $N(0, 1)$ case.
- Changes occurring right after time-point $\tau = 300$ are easier to detect compared to the changes occurring right after $\tau = 50$. MW, however, cannot detect a decrease in scale at all.

19.4.3 Location-Scale Changes

Now, we focus on changes in location and scale simultaneously. We consider 8 changes in location and scale simultaneously, namely, $(\delta_L, \delta_S) = (0.50, 1.50), (0.50, 0.50), (0.50, 2.00), (0.50, 0.25), (1.00, 1.50), (1.00, 0.50), (1.00, 2.00),$ and $(1.00, 0.25)$. For each of the three distributions $N(0, 1), ST(2.5),$ and $SLN(1.0, 0.5)$, we generate 50,000 sequences of observations where the post-change observations are calculated by multiplying δ_S and then adding δ_L . From Tables 19.3, 19.4, and 19.5, we observe that

19.4.4 General Distributional Changes

Finally, we consider general changes of various “in-control” distributions. The corresponding changes in the pair of mean and standard deviations are large in some cases (e.g., Weibull(1) to Weibull(3) and vice versa, Gamma(2,2) to Gamma(3,2) and vice versa) and small or zero in some other cases (e.g., $N(0, 1)$ to $ST(2.5)$ and vice versa, $N(0, 1)$ to $SLN(1.0, 0.5)$ and vice versa). From Table 19.6, we observe that

Table 19.3 Mean delay in detection of various amounts of changes in location and scale simultaneously, i.e., (δ_L, δ_S) , occurring right after time τ . “In-control” distribution considered here is $N(0, 1)$. The results are based on 50,000 random simulations when $ARL_0 = 500$

$N(\delta_L, \delta_S^2)$	(.5, 1.5)	(.5, .5)	(.5, 2)	(.5, .25)	(1, 1.5)	(1, .5)	(1, 2)	(1, .25)
$\tau = 50$ and $(\delta_L, \delta_S) =$								
MW	137.1	125.4	127.2	129.0	20.2	9.4	28.3	8.3
Mood	116.6	45.6	18.2	20.4	168.0	62.1	23.3	44.5
Lepage	90.4	31.2	20.7	16.8	19.7	14.0	12.5	12.0
CvM	131.0	66.9	94.6	19.9	20.9	8.7	24.0	6.6
Proposed	116.4	33.1	33.5	14.0	21.7	9.2	19.0	7.1
$\tau = 300$ and $(\delta_L, \delta_S) =$								
$N(\delta_L, \delta_S^2)$	(.5, 1.5)	(.5, .5)	(.5, 2)	(.5, .25)	(1, 1.5)	(1, .5)	(1, 2)	(1, .25)
MW	38.7	26.9	37.1	23.7	13.6	7.5	16.4	6.6
Mood	22.5	52.0	9.9	26.9	14.0	148.4	8.3	136.8
Lepage	21.1	27.7	11.3	18.9	10.2	10.7	8.3	9.9
CvM	37.9	20.7	30.4	13.1	13.9	7.0	15.6	5.6
Proposed	33.9	20.8	20.8	13.8	15.1	8.5	14.6	7.0

Table 19.4 Mean delay in detection of various amounts of changes in location and scale simultaneously, i.e., (δ_L, δ_S) , occurring right after time τ . “In-control” distribution considered here is $ST(2.5)$. The results are based on 50,000 random simulations when $ARL_0 = 500$

$ST(2.5) \cdot \delta_S + \delta_L$	(.5, 1.5)	(.5, .5)	(.5, 2)	(.5, .25)	(1, 1.5)	(1, .5)	(1, 2)	(1, .25)
$\tau = 50$ and $(\delta_L, \delta_S) =$								
MW	33.9	13.4	47.4	10.8	7.7	4.7	10.0	4.3
Mood	241.9	66.4	70.1	41.8	110.2	52.5	87.0	46.3
Lepage	57.1	18.5	34.1	14.2	7.4	5.1	7.9	4.7
CvM	31.4	11.0	36.7	7.9	7.4	4.3	9.3	3.8
Proposed	33.3	11.6	30.6	8.3	7.9	4.7	9.3	4.2
$\tau = 300$ and $(\delta_L, \delta_S) =$								
MW	17.3	9.4	20.5	7.9	6.4	4.1	7.9	3.8
Mood	25.0	166.7	12.5	120.7	7.6	6.9	6.8	6.0
Lepage	14.7	14.6	11.6	12.7	5.1	3.7	5.4	3.4
CvM	16.8	8.5	18.7	6.5	6.2	3.8	7.5	3.4
Proposed	17.8	9.9	17.3	7.9	7.7	5.3	8.7	4.8

Table 19.5 Mean delay in detection of various amounts of changes in location and scale simultaneously, i.e., (δ_L, δ_S) , occurring right after time τ . “In-control” distribution considered here is $SLN(1.0, 0.5)$. The results are based on 50,000 random simulations when $ARL_0 = 500$

$(SLN(1, .5)).\delta_S + \delta_L$	(.5, 1.5)	(.5, .5)	(.5, 2)	(.5, .25)	(1, 1.5)	(1, .5)	(1, 2)	(1, .25)
$\tau = 50$ and $(\delta_L, \delta_S) =$								
MW	201.5	40.7	216.0	33.9	19.8	7.9	44.2	7.4
Mood	273.6	32.2	28.5	24.9	384.2	47.4	140.2	42.5
Lepage	217.9	19.1	45.0	15.2	33.3	11.7	48.1	10.7
CvM	206.1	17.9	185.3	11.1	21.6	6.6	42.2	5.6
Proposed	224.4	14.8	67.7	10.5	23.8	7.0	44.7	6.1
$\tau = 300$ and $(\delta_L, \delta_S) =$								
MW	53.3	15.1	65.2	13.0	12.9	6.4	20.2	5.9
Mood	69.0	55.8	13.3	48.3	154.7	137.8	18.9	133.5
Lepage	44.7	19.8	16.1	17.5	15.7	9.3	14.6	8.6
CvM	54.4	11.6	47.8	8.6	13.6	5.5	20.1	4.9
Proposed	55.7	12.6	28.4	9.8	15.3	7.0	20.6	6.3

Table 19.6 Mean delay in detection of various general changes in distribution. The results are based on 50,000 random simulations when $ARL_0 = 500$

	MW	Mood	Lepage	CvM	P_{250}	P_{500}	P_{1000}	P_∞
$\tau = 50$								
$N(0, 1) \rightarrow ST(2.5)$	685.7	109.4	192.2	617.5	173.1	178.1	173.4	173.1
$ST(2.5) \rightarrow N(0, 1)$	299.6	101.5	149.7	272.3	125.8	109.9	100.8	96.1
$N(0, 1) \rightarrow SLN(1, 0.5)$	555.5	539.1	544.9	516.2	455.8	478.0	487.8	499.1
$SLN(1, 0.5) \rightarrow N(0, 1)$	430.7	376.9	366.8	416.7	423.0	418.8	412.5	412.5
$\text{Gamma}(2,2) \rightarrow \text{Gamma}(3,2)$	46.8	466.0	86.3	55.0	63.0	60.2	60.5	62.2
$\text{Gamma}(3,2) \rightarrow \text{Gamma}(2,2)$	39.9	464.5	76.8	46.4	56.6	52.9	51.4	52.2
$\text{Weibull}(1) \rightarrow \text{Weibull}(3)$	657.5	18.3	23.2	177.7	25.5	24.9	24.0	23.6
$\text{Weibull}(3) \rightarrow \text{Weibull}(1)$	117.8	8.0	9.7	52.7	15.7	15.2	14.5	14.2
$\text{Uniform}(0,1) \rightarrow \text{Beta}(5,5)$	794.2	19.5	28.5	385.3	34.5	34.4	32.5	32.4
$\text{Beta}(5,5) \rightarrow \text{Uniform}(0,1)$	178.5	9.7	12.5	119.6	20.1	19.4	18.5	17.9
$\tau = 300$								
$N(0, 1) \rightarrow ST(2.5)$	1168.9	37.4	55.9	190.5	51.3	47.2	46.2	45.7
$ST(2.5) \rightarrow N(0, 1)$	129.9	23.0	28.7	79.4	40.6	37.4	36.3	35.9
$N(0, 1) \rightarrow SLN(1, 0.5)$	643.8	462.1	458.5	449.7	356.8	346.7	373.0	419.4
$SLN(1, 0.5) \rightarrow N(0, 1)$	316.1	168.9	178.0	265.0	322.5	277.9	269.6	288.0
$\text{Gamma}(2,2) \rightarrow \text{Gamma}(3,2)$	19.7	316.8	27.4	21.0	23.8	22.9	22.4	22.2
$\text{Gamma}(3,2) \rightarrow \text{Gamma}(2,2)$	18.7	155.3	21.7	20.2	22.9	22.2	21.8	21.5
$\text{Weibull}(1) \rightarrow \text{Weibull}(3)$	512.2	15.7	20.7	37.9	19.7	20.3	19.8	19.6
$\text{Weibull}(3) \rightarrow \text{Weibull}(1)$	31.1	5.9	7.1	21.8	13.4	14.2	14.0	13.8
$\text{Uniform}(0,1) \rightarrow \text{Beta}(5,5)$	1670.0	15.2	22.1	60.1	21.3	22.1	21.7	21.4
$\text{Beta}(5,5) \rightarrow \text{Uniform}(0,1)$	48.9	7.0	8.7	30.2	16.5	17.3	16.8	16.6

- No one method is uniformly best. P_{500} works well in all cases. Note that P_{500} is the proposed method when $T_0 = 500$ as defined in the fourth paragraph of Sect. 19.3.
- In cases where another method is the best, P_{500} is not far behind.
- The performance of P_{500} is not far from the best choice of T_0 for the proposed method. The larger the value of T_0 , the more the computing time. Therefore, P_{500} is a good balance between computing time and performance. Hence, we suggest using $T_0 = ARL_0$ in most applications.
- In cases where location change is large, MW works well, and when scale change is large, Mood works well, as expected.
- Distributional change from $N(0, 1)$ to $SLN(1.0, 0.5)$ which does not alter the mean and standard deviation is detected by the proposed method earlier than other charts.
- When we are not sure about the nature of possible distributional change, the proposed method is a good choice.
- Changes occurring right after time-point $\tau = 300$ are easier to detect compared to the changes occurring right after $\tau = 50$. MW cannot detect an arbitrary distributional change well if the median does not change by much.

Table 19.7 Mean delay in detection of various location shifts δ_L occurring right after time τ . “In-control” distributions considered are $N(0, 1)$, $ST(2.5)$, and $SLN(1.0, 0.5)$. The results are based on 10,000 random simulations when $ARL_0 = 200$

	$\tau = 30$				$\tau = 150$			
	$\delta_L =$				$\delta_L =$			
$N(\delta_L, 1)$	0.25	0.50	1.00	2.00	0.25	0.50	1.00	2.00
MW	165.4	85.4	13.3	3.8	102.5	29.2	8.5	3.2
Mood	195.3	198.3	156.4	36.9	185.2	161.5	54.4	2.6
Lepage	150.6	100.4	18.9	3.1	112.7	39.0	9.2	2.3
CvM	166.4	95.1	15.2	3.7	106.6	32.5	8.9	3.1
Proposed	166.5	103.0	16.8	4.3	116.0	37.6	10.2	4.0
<i>ST(2.5) + δ_L</i>								
MW	110.0	24.3	5.2	3.0	40.0	11.1	4.2	2.7
Mood	191.0	154.4	52.6	17.4	178.8	105.4	8.8	2.0
Lepage	118.5	35.7	6.1	1.9	57.5	14.2	3.8	1.5
CvM	110.5	22.8	5.0	2.6	40.2	10.9	4.0	2.3
Proposed	119.1	26.1	5.6	3.2	46.6	12.2	4.9	3.1
<i>SLN(1.0, 0.5) + δ_L</i>								
MW	167.9	78.9	11.0	3.7	94.0	22.9	7.0	3.1
Mood	205.1	150.6	52.7	34.8	202.9	106.1	76.1	6.0
Lepage	153.0	69.9	13.6	4.1	101.8	30.1	9.8	2.7
CvM	165.8	82.5	10.6	3.4	95.0	23.5	7.0	2.9
Proposed	158.7	72.8	11.2	4.0	89.5	24.8	7.9	3.7

Table 19.8 Mean delay in detection of various scale changes δ_S occurring right after time τ . “In-control” distributions considered are $N(0, 1)$, $ST(2.5)$, and $SLN(1.0, 0.5)$. The results are based on 10,000 random simulations when $ARL_0 = 200$

	$\tau = 30$				$\tau = 150$			
	$\delta_S =$				$\delta_S =$			
$N(0, \delta_S^2)$	1.50	2.00	0.50	0.25	1.50	2.00	0.50	0.25
MW	129.2	93.5	299.8	340.8	73.4	39.8	561.6	739.2
Mood	60.7	16.8	39.8	9.6	20.1	8.0	17.5	8.1
Lepage	61.8	20.4	61.8	13.1	22.7	9.2	25.8	11.9
CvM	128.0	87.7	253.5	97.2	68.2	31.5	106.2	26.9
Proposed	94.3	31.6	70.2	13.7	45.7	18.9	26.9	10.9
<i>ST(2.5).δ_S</i>								
MW	145.2	111.4	284.3	332.1	92.2	54.1	481.6	685.7
Mood	103.4	41.3	68.7	12.4	37.6	13.8	25.3	9.9
Lepage	90.7	46.1	97.4	17.3	41.9	16.0	38.4	14.3
CvM	144.2	105.1	256.2	151.4	88.0	44.1	173.5	34.5
Proposed	119.5	55.5	95.9	18.3	68.0	26.0	35.3	13.2
<i>SLN(1.0, 0.5).δ_S</i>								
MW	113.8	75.2	281.6	296.5	55.7	29.0	381.0	357.7
Mood	47.6	13.9	31.9	10.1	15.6	6.6	16.4	9.1
Lepage	46.3	13.9	43.8	12.8	16.9	7.4	22.2	12.2
CvM	110.3	65.5	217.7	69.3	51.9	22.7	64.0	21.0
Proposed	76.1	22.8	53.8	12.3	34.8	15.5	22.5	10.6

Numerical simulations when $ARL_0 = 200$ are provided in Tables 19.7, 19.8, 19.9, 19.10, 19.11, and 19.12. Similar conclusions as provided above can be drawn from these tables as well.

From these simulation studies, we see that the proposed method works well in most applications. When we are trying to detect a specific type of distributional change, its performance is slightly worse than the chart that is designed to detect that particular type of changes. However, in most cases, the differences are not much. Therefore,

when the nature of distributional change is unknown, the proposed chart can be used with anticipation of a good performance.

19.5 Analysis of Various Real-World Data

Now, we focus on applications of the proposed chart on real-world problems. We consider two datasets: a climate data and a blood glucose monitoring data.

Table 19.9 Mean delay in detection of various changes in location and scale simultaneously, i.e., (δ_L, δ_S) occurring right after time τ . “In-control” distribution considered here is $N(0, 1)$. The results are based on 10,000 random simulations when $ARL_0 = 200$

$N(\delta_L, \delta_S^2)$	(.5, 1.5)	(.5, .5)	(.5, 2)	(.5, .25)	(1, 1.5)	(1, .5)	(1, 2)	(1, .25)
$\tau = 30$ and $(\delta_L, \delta_S) =$								
MW	76.2	99.7	68.0	103.7	19.4	9.5	24.4	8.7
Mood	71.2	41.9	18.4	15.9	93.2	42.9	24.4	29.9
Lepage	45.7	29.2	17.2	13.3	17.8	12.0	11.0	9.8
CvM	75.8	71.1	59.2	26.4	20.2	8.4	22.2	6.2
Proposed	70.3	33.0	28.2	11.9	22.3	9.0	17.9	6.6
$\tau = 150$ and $(\delta_L, \delta_S) =$								
MW	29.3	24.7	25.0	22.2	10.9	6.2	12.6	5.5
Mood	17.5	37.6	7.5	20.0	11.4	82.8	6.5	76.1
Lepage	15.2	21.8	8.1	14.7	7.8	8.8	6.0	8.3
CvM	28.8	18.3	21.7	11.4	11.1	5.8	11.8	4.7
Proposed	27.0	17.6	16.3	11.3	12.2	6.9	11.4	5.5

Table 19.10 Mean delay in detection of various changes in location and scale simultaneously, i.e., (δ_L, δ_S) occurring right after time τ . “In-control” distribution considered here is $ST(2.5)$. The results are based on 10,000 random simulations when $ARL_0 = 200$

$ST(2.5).\delta_S + \delta_L$	(.5, 1.5)	(.5, .5)	(.5, 2)	(.5, .25)	(1, 1.5)	(1, .5)	(1, 2)	(1, .25)
$\tau = 30$ and $(\delta_L, \delta_S) =$								
MW	32.4	15.1	38.8	12.7	6.9	4.1	8.9	3.8
Mood	111.1	47.4	51.9	28.6	74.8	36.9	59.3	32.0
Lepage	42.1	15.4	28.2	11.3	7.1	4.9	7.9	4.4
CvM	30.4	11.8	32.8	7.5	6.5	3.7	8.2	3.3
Proposed	34.5	11.5	29.5	7.7	7.2	4.3	8.4	3.8
$\tau = 150$ and $(\delta_L, \delta_S) =$								
MW	14.0	7.9	15.8	6.7	5.4	3.4	6.4	3.1
Mood	24.2	91.7	10.6	68.9	6.9	10.6	5.8	10.1
Lepage	12.0	12.1	9.2	10.5	4.3	3.1	4.3	2.8
CvM	13.4	7.1	14.4	5.4	5.1	3.1	6.2	2.8
Proposed	14.4	8.2	13.7	6.3	6.0	3.9	6.8	3.5

Table 19.11 Mean delay in detection of various changes in location and scale simultaneously, i.e., (δ_L, δ_S) occurring right after time τ . “In-control” distribution considered here is $SLN(1.0, 0.5)$. The results are based on 10,000 random simulations when $ARL_0 = 200$

$SLN(1, 0.5).\delta_S + \delta_L$	(.5, 1.5)	(.5, .5)	(.5, 2)	(.5, .25)	(1, 1.5)	(1, .5)	(1, 2)	(1, .25)
$\tau = 30$ and $(\delta_L, \delta_S) =$								
MW	101.6	49.1	96.5	43.4	20.7	8.0	38.1	7.3
Mood	119.9	24.2	26.0	18.1	176.4	31.9	76.3	27.8
Lepage	85.2	15.5	28.5	11.8	30.1	9.7	34.1	8.9
CvM	105.9	24.1	84.9	11.4	23.1	6.2	37.7	5.1
Proposed	110.3	13.6	47.4	9.0	27.6	6.5	40.9	5.6
$\tau = 150$ and $(\delta_L, \delta_S) =$								
MW	40.6	13.2	38.6	11.3	10.6	5.3	16.2	4.9
Mood	47.9	37.0	10.2	32.1	91.3	76.2	15.6	74.7
Lepage	31.8	15.6	11.0	13.6	12.8	7.8	11.2	7.2
CvM	42.5	9.9	32.5	7.3	11.3	4.6	16.2	4.0
Proposed	47.6	10.5	21.9	8.1	13.0	5.5	16.5	4.8

19.5.1 Climate Data on Minneapolis, USA

Monitoring for the changes of patterns of various climatological measurements such as maximum and minimum temperatures on a daily, monthly, and yearly basis, amounts of rainfall, snow, measurements of snow depth, number of rainy or snowy days is an emerging research area. In the literature, various statistical methods are demonstrated with the capacity of analyzing such data. Modern statistical process control (SPC) charts also deserve a chance to monitor such climatological variables. In this context, we consider

mean daily maximum temperature in Fahrenheit in the month of January in Minneapolis, USA. We collect the data from http://www.dnr.state.mn.us/climate/twin_cities/listings.html. The data are from 1873 to 2017 with no missing value and are presented in Fig. 19.1. At first, before applying the control charts, we check if our assumption of independence of the observations is reasonable. Durbin-Watson test (R function: *dwtest*, R package: *lmtest*) for two-sided alternative gives a high p-value 0.8359 showing lack of autocorrelation. Therefore, we can assume that the observations are independent. Now, we apply the

Table 19.12 Mean delay in detection of various general changes in distribution. The results are based on 10,000 random simulations when $ARL_0 = 200$

	MW	Mood	Lepage	CvM	P_{100}	P_{200}	P_{400}	P_{∞}
$\tau = 30$								
$N(0, 1) \rightarrow ST(2.5)$	276.8	79.9	107.6	256.3	104.8	105.6	109.4	113.2
$ST(2.5) \rightarrow N(0, 1)$	121.0	67.0	70.2	114.0	78.5	69.1	63.5	62.1
$N(0, 1) \rightarrow SLN(1, 0.5)$	218.2	210.8	176.9	206.8	173.4	183.4	186.7	192.0
$SLN(1, 0.5) \rightarrow N(0, 1)$	174.1	155.1	121.3	166.3	161.7	160.5	159.0	157.2
$\text{Gamma}(2,2) \rightarrow \text{Gamma}(3,2)$	45.2	191.7	59.2	50.2	58.4	56.7	58.6	55.9
$\text{Gamma}(3,2) \rightarrow \text{Gamma}(2,2)$	38.4	184.8	51.3	42.4	54.6	50.3	49.0	48.8
$\text{Weibull}(1) \rightarrow \text{Weibull}(3)$	286.1	16.9	21.6	153.6	27.3	26.4	25.9	26.0
$\text{Weibull}(3) \rightarrow \text{Weibull}(1)$	56.8	7.7	7.6	41.2	14.0	12.7	12.4	12.0
$\text{Uniform}(0,1) \rightarrow \text{Beta}(5,5)$	322.6	19.7	29.3	219.5	38.8	37.7	39.4	41.4
$\text{Beta}(5,5) \rightarrow \text{Uniform}(0,1)$	77.9	9.2	9.7	62.8	17.9	16.4	15.4	15.0
$\tau = 150$								
$N(0, 1) \rightarrow ST(2.5)$	444.9	30.4	45.0	194.6	60.6	40.7	40.7	39.7
$ST(2.5) \rightarrow N(0, 1)$	61.9	19.0	22.0	51.6	44.6	29.0	28.4	28.2
$N(0, 1) \rightarrow SLN(1, 0.5)$	252.4	202.7	168.9	209.6	156.4	157.2	174.9	196.9
$SLN(1, 0.5) \rightarrow N(0, 1)$	137.9	85.4	69.6	123.6	141.3	127.8	129.7	131.0
$\text{Gamma}(2,2) \rightarrow \text{Gamma}(3,2)$	16.4	151.0	22.2	17.7	23.7	20.0	19.3	18.8
$\text{Gamma}(3,2) \rightarrow \text{Gamma}(2,2)$	15.5	86.6	17.0	16.6	22.1	19.1	18.3	17.9
$\text{Weibull}(1) \rightarrow \text{Weibull}(3)$	340.9	12.7	16.7	34.2	17.6	16.6	16.0	16.1
$\text{Weibull}(3) \rightarrow \text{Weibull}(1)$	20.8	4.5	5.0	15.4	11.5	11.0	10.7	10.4
$\text{Uniform}(0,1) \rightarrow \text{Beta}(5,5)$	640.5	12.4	18.0	56.2	21.6	18.5	18.2	18.0
$\text{Beta}(5,5) \rightarrow \text{Uniform}(0,1)$	29.3	5.4	6.2	20.4	13.9	13.2	12.8	12.6

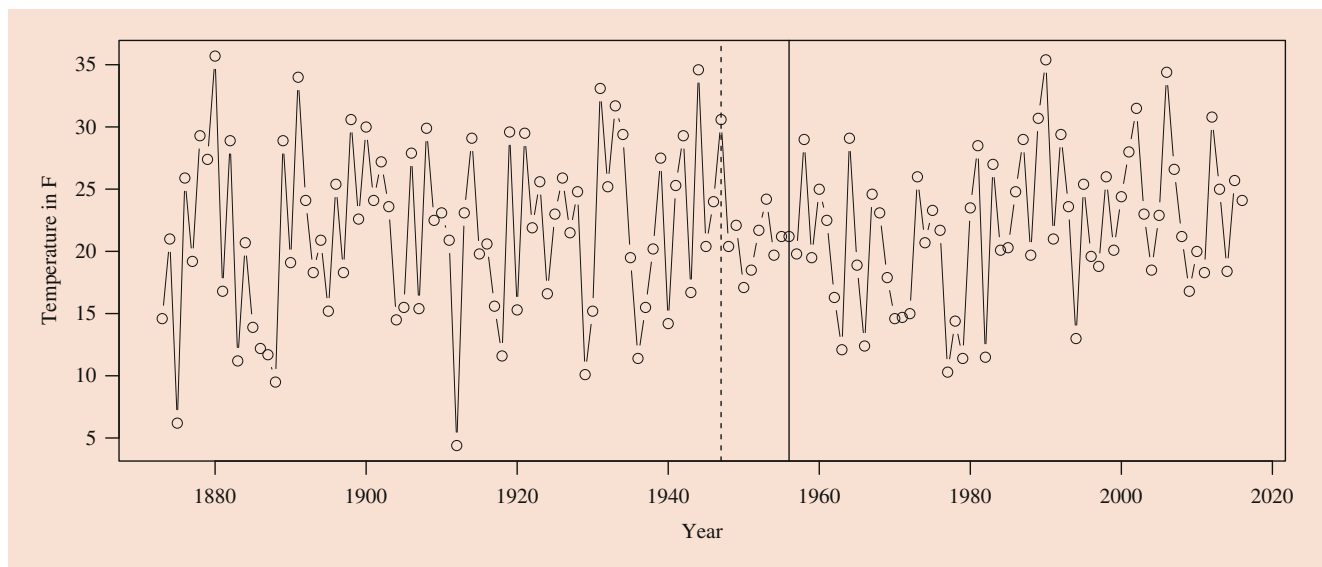


Fig. 19.1 Mean daily maximum temperature in the month of January in Minneapolis, USA. Setting $ARL_0 = 500$, the proposed chart detects a distributional change in the year 1956 estimating the change-point to be just after 1947. Other charts do not detect any distributional change

proposed method along with other competing methods when we set $ARL_0 = 500$. For the proposed method, we set $T_0 = ARL_0 = 500$ as per our suggestion mentioned before. The proposed method detects a distributional change at 1956 while the estimated change-point is 1947. All other methods, i.e., MW, Mood, Lepage, and CvM, do not detect any distributional change. We also run all SPC charts when we set $ARL_0 = 200$. In this case also, the proposed method detects a distributional change at 1956 while the estimated change-point is 1947. Mood also detects distributional change but at

a later time at 1960 while the estimated change point is 1947. MW, Lepage, and CvM still do not detect any distributional change. Now, we study the “in-control” distribution and the estimated “out-of-control” distribution. Table 19.13 shows the first four sample moments of “in-control” observations from 1873 to 1947 and “out-of-control” observations from 1948 to 1956. From Table 19.13, we see that the second moment changed a lot, but not the other three moments. We carry out the calculations using R-package *moments*.

19.5.2 Blood Glucose Monitoring Data

Monitoring blood glucose level on a daily basis is essential for advanced diabetic patients. It gives information on whether the particular lifestyle change and the treatment procedure including the medicine with its administered dosage are working well for the patient. Monitoring such measurements is complicated because we are not just focusing on the mean and the standard deviation only; we need to monitor the stability of its probability distribution as well. We collect a data from UCI Machine Learning Repository [77]. This directory contains a dataset prepared for the use of participants for the 1994 AAAI Spring Symposium on Artificial Intelligence in Medicine. For our analysis, we pick the data from the first patient. We choose to monitor pre-breakfast blood glucose level on a daily basis. We prefer this over blood glucose measurements at other times because they depend too much on many variables like the type of food, amount of food, etc. The data from the first patient contains daily pre-breakfast observations from 21 April 1991 to 3 September 1991 with the number on 18 August 1991 missing. The data are presented in Fig. 19.2 ignoring the only one missing value. As always, before applying the control chart, we check if our assumption of independence of the observations is reasonable. Durbin-Watson test for two-sided

alternative gives a high p-value 0.7727 showing lack of autocorrelation. Therefore, our assumption of independence is reasonable. Now, we apply the proposed method along with other competing methods when we set $ARL_0 = 500$. For the proposed method, we set $T_0 = ARL_0 = 500$ like before. The proposed method detects a distributional change on 11 June 1991 while the estimated change-point is on 5 June 1991. Other competing charts do not detect any distributional change. Running the charts for $ARL_0 = 200$ gives similar results except that the proposed chart detects distributional change one day earlier on 10 June 1991 while the estimated change-point being the same as before. Table 19.14 shows first four moments before and after the distributional change. The standard deviation decreases a lot showing a more stable fasting blood glucose numbers, and also the skewness appears to have decreased considerably. It is to be noted that the phrases “in-control” and “out-of-control” we use are in the sense of standard terminologies in SPC literature, not in the sense of blood glucose control.

Table 19.13 First four sample moments of the observations from 1873 to 1947 and 1948 to 1956

	Mean	St. deviation	Skewness	Kurtosis
1873–1947, “in-control”	21.7067	7.0842	−0.1898	2.3666
1948–1956, “out-of-control”	20.6778	2.0795	−0.1350	2.6056

19.6 Concluding Remarks

This chapter first describes a few commonly used traditional statistical process control (SPC) charts such as Shewhart, CUSUM, EWMA, and change-point-based (CPD) control charts and discusses a number of situations where these charts should be appropriately modified for practical use. Next, this chapter proposes a change-point based nonparametric statistical process control chart for detecting arbitrary distributional changes when the process distribution is univariate continuous. There are two specifically important

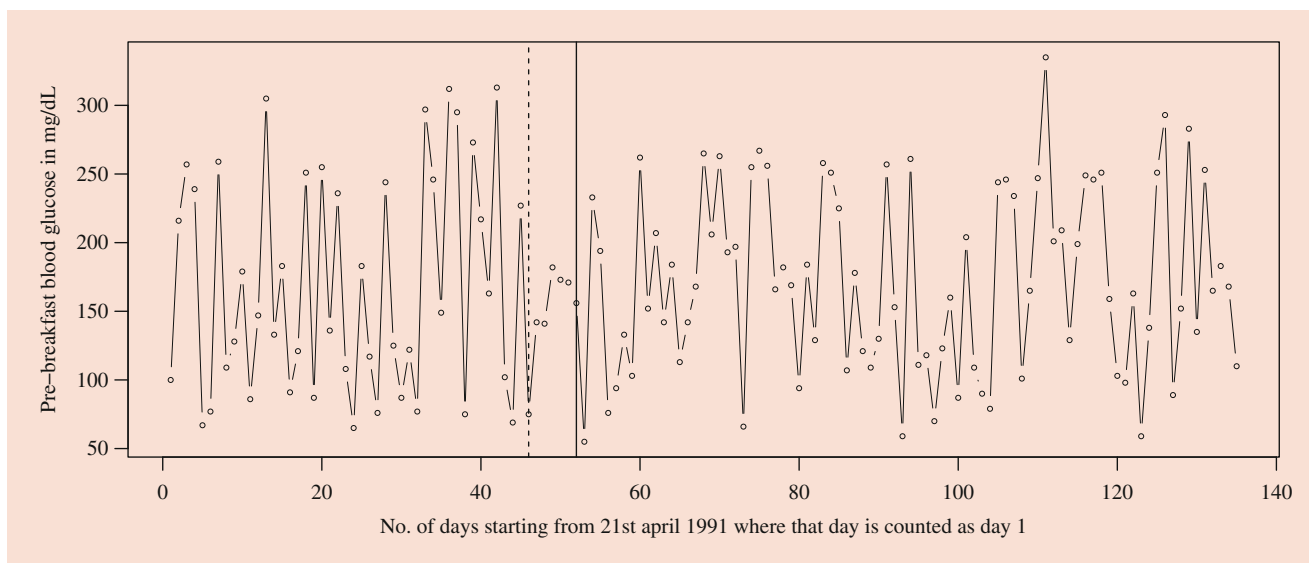


Fig. 19.2 Daily pre-breakfast blood glucose measurements of a selected patient. Setting $ARL_0 = 500$, the proposed chart detects a distributional change at the 52nd day estimating the change-point to be the 46th day. Other charts do not detect any distributional change

Table 19.14 First four sample moments of the observations from 21 April 1991 to 5–6 June 1991 to 11 June 1991

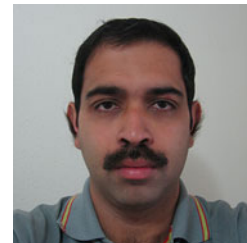
	Mean	St. deviation	Skewness	Kurtosis
21st April to 5th June, “in-control”	166.9	80.8	0.3824	1.7221
6th June to 11th June, “out-of-control”	160.8	17.2	−0.0981	1.4267

contributions of the proposed chart. The first one is the combination of the strengths of two statistical tests: Cramer-von Mises test and Ansari-Bradley test. The second one is the introduction of a numerically efficient technique to estimate the possible change-point without sacrificing the accuracy by much. Both these contributions are quite general in nature and have broad applications well beyond the numerical examples and real-world data analyses shown in this chapter including monitoring fast data streams. Another important aspect of this control chart is that runtime distribution is not geometric even when the process is “in-control.” The reason is that the probability of getting a signal is a function of current runtime. However, it appears that such charts can still be used in many applications. However, further research is required to fully understand the pros and cons of such charts.

References

- Shewhart, W.A.: *Economic Control of Quality of Manufactured Product*. Van Nostrand, New York (1931)
- Page, E.S.: Continuous inspection schemes. *Biometrika* **41**, 100–114 (1954)
- Borror, C.M., Montgomery, D.C., Runger, G.C.: Robustness of the EWMA control chart to non-normality. *J. Qual. Technol.* **31**, 309–316 (1999)
- Hawkins, D.M., P. Qiu, Kang, C.W.: The changepoint model for statistical process control. *J. Qual. Technol.* **35**, 355–366 (2003)
- Zhou, C., Zou, C., Zhang, Y., Wang, Z.: Nonparametric control chart based on change-point model. *Stat. Pap.* **50**, 13–28 (2009)
- Amin, R., M.R. Reynolds Jr., Bakir, S.T.: Nonparametric quality control charts based on the sign statistic. *Commun. Stat. Theory Methods* **24**, 1597–1623 (1995)
- Hackl, P., Ledolter, J.: A new nonparametric quality control technique. *Commun. Stat. Simul. Comput.* **21**, 423–443 (1992)
- Albers, W., Kallenberg, W.C.M.: Empirical nonparametric control charts: estimation effects and corrections. *J. Appl. Stat.* **31**, 345–360 (2004)
- Albers, W., Kallenberg, W.C.M.: CUMIN charts. *Metrika* **70**, 111–130 (2009)
- Chakraborti, S., van der Laan, P., van de Wiel, M.A.: A class of distribution-free control charts. *J. R. Stat. Soc. Ser. C Appl. Stat.* **53**, 443–462 (2004)
- Chakraborti, S., Eryilmaz, S., Human, S.W.: A phase II nonparametric control chart based on precedence statistics with runs-type signaling rules. *Comput. Stat. Data Anal.* **53**, 1054–1065 (2009)
- Chakraborti, S., Eryilmaz, S.: A nonparametric Shewhart-type signed-rank control chart based on runs. *Commun. Stat. Simul. Comput.* **36**, 335–356 (2007)
- Hawkins, D.M., Deng, Q.: A nonparametric change-point control chart. *J. Qual. Technol.* **42**, 165–173 (2010)
- Liu, L., Tsung, F., Zhang, J.: Adaptive nonparametric CUSUM scheme for detecting unknown shifts in location. *Int. J. Prod. Res.* **52**, 1592–1606 (2014)
- Mukherjee, A., Chakraborti, S.: A distribution-free control chart for the joint monitoring of location and scale. *Qual. Reliab. Eng. Int.* **28**(3), 335–352 (2012)
- Qiu, P., Li, Z.: On nonparametric statistical process control of univariate processes. *Technometrics* **53**, 390–405 (2011)
- Qiu, P., Li, Z.: Distribution-free monitoring of univariate processes. *Stat. Prob. Lett.* **81**, 1833–1840 (2011)
- Qiu, P., Zhang, J.: On phase II SPC in cases when normality is invalid. *Qual. Reliab. Eng. Int.* **31**, 27–35 (2015)
- Qiu, P.: Some perspectives on nonparametric statistical process control. *J. Qual. Technol.* **50**, 49–65 (2018)
- Qiu, P., Hawkins, D.M.: A rank based multivariate CUSUM procedure. *Technometrics* **43**, 120–132 (2001)
- Qiu, P., Hawkins, D.M.: A nonparametric multivariate cumulative sum procedure for detecting shifts in all directions. *J. R. Stat. Soc. Ser. D Stat.* **52**, 151–164 (2003)
- Qiu, P.: Distribution-free multivariate process control based on log-linear modeling. *IIE Trans.* **40**, 664–677 (2008)
- Yeh, A.B., Lin, D.K.J., Venkatramani, C.: Unified CUSUM charts for monitoring process mean and variability. *Qual. Technol. Quant. Manag.* **1**, 65–86 (2004)
- Hawkins, D.M., Zamba, K.D.: Statistical process control for shifts in mean or variance using a changepoint formulation. *Technometrics* **47**, 164–173 (2005)
- Zhang, J., Li, E., Li, Z.: A Cramer-von Mises test-based distribution-free control chart for joint monitoring of location and scale. *Comput. Indust. Eng.* **110**, 484–497 (2017)
- Moustakides, G.V.: Optimal stopping times for detecting changes in distributions. *Ann. Stat.* **14**, 1379–1387 (1986)
- Ross, G.J., Adams, N.M.: Two nonparametric control charts for detecting arbitrary distribution changes. *J. Qual. Technol.* **44**, 102–116 (2012)
- Mukherjee, P.S.: On phase II monitoring of the probability distributions of univariate continuous processes. *Stat. Pap.* **57**(2), 539–562 (2016)
- Hawkins, D.M., Olwell, D.H.: *Cumulative Sum Charts and Charting for Quality Improvement*. Springer, New York (1998)
- Qiu, P.: *Introduction to Statistical Process Control*. CRC Press, Taylor & Francis Group, New York (2013)
- Li, Z., Qiu, P., Chatterjee, S., Wang, Z.: Using p-values to design statistical process control charts. *Stat. Pap.* **54**, 523–539 (2013)
- Li, Z., Qiu, P.: Statistical process control using dynamic sampling. *Technometrics* **56**, 325–335 (2014)
- Ross, G.J., Tasoulis, D.K., Adams, N.M.: Nonparametric monitoring of data streams for changes in location and scale. *Technometrics* **53**(4), 379–389 (2011)
- Kenney, J.F., Keeping, E.S.: *Mathematics of Statistics, Part Twol*. Van Nostrand Company Inc., Princeton (1951)
- Chou, Y.M., Polansky, A.M., Mason, R.L.: Transforming non-normal data to normality in statistical process control. *J. Qual. Technol.* **30**, 133–141 (1998)
- Yourstone, S.A., Zimmer, W.J.: Non-normality and the design of control charts for averages. *Decis. Sci.* **23**, 1099–1113 (1992)
- Alwan, L.C.: Effects of autocorrelation on control chart performance. *Commun. Stat. Theory Methods* **21**, 1025–1049 (1992)
- Black, R., Smith, J., Wells, S.: The impact of Weibull data and autocorrelation on the performance of the Shewhart and exponentially weighted moving average control charts. *Int. J. Ind. Eng. Comput.* **2**, 575–582 (2011)

39. English, J.R., Lee, S.C., Martin, T.W., Tilmon, C.: Detecting changes in autoregressive processes with X and EWMA charts. *IIE Tras.* **32**, 1103–1113 (2000)
40. Runger, G.C., Willemains, T.R.: Batch means control charts for autocorrelated data. *IIE Tras.* **28**, 483–487 (1996)
41. Kim, S.H., Alexopoulos, C., Tsui, K.L., Wilson, J.R.: A distribution-free tabular CUSUM chart for autocorrelated data. *IIE Tras.* **39**, 317–330 (2007)
42. Runger, G.C., Willemains, T.R.: Model-based and model-free control of autocorrelated processes. *J. Qual. Technol.* **27**, 283–292 (1995)
43. Johnson, R.A., Bagshaw, M.: The effect of serial correlation on the performance of CUSUM tests. *Technometrics* **16**, 103–112 (1974)
44. Montgomery, D.C., Mastrangelo, C.M.: Some statistical process control methods for autocorrelated data. *J. Qual. Technol.* **23**, 179–204 (1991)
45. Jiang, W., Tsui, K.L., Woodall, W.: A new SPC monitoring method: the ARMA chart. *Technometrics* **42**, 399–410 (2000)
46. Hawkins, D.M.: Self-starting cusums for location and scale. *Statistician* **36**, 299–315 (1987)
47. Hawkins, D.M.: On the distribution and power of a test for a single outlier. *S. Afr. Stat. J.* **3**, 9–15 (1969)
48. Chatterjee, S., Qiu, P.: Distribution-free cumulative sum control charts using bootstrap-based control limits. *Ann. Appl. Stat.* **3**, 349–369 (2009)
49. Hawkins, D.M., Maboudou-Tchao, E.M.: Self-starting multivariate exponentially weighted moving average control charting. *Technometrics* **49**, 199–209 (2007)
50. Sparks, R.S.: CUSUM charts for signalling varying location shifts. *J. Qual. Technol.* **32**, 157–171 (2000)
51. Gan, F.F.: CUSUM control charts under linear drift. *Statistician* **41**, 71–84 (1992)
52. Davis, R.B., Woodall, W.H.: Performance of the control chart trend rule under linear drift. *J. Qual. Technol.* **20**, 260–262 (1988)
53. Costa, A.F.B.: Joint \bar{X} and R charts with variable parameters. *IIE Trans.* **30**, 505–514 (1998)
54. Luo, Y., Li, Z., Wang, Z.: AdaptiveCUSUM control chart with variable sampling intervals. *Comput. Stat. Data Anal.* **53**, 2693–2701 (2009)
55. Roberts, S.V.: Control chart tests based on geometric moving averages. *Technometrics* **1**, 239–250 (1959)
56. Crowder, S.V.: Design of exponentially weighted moving average schemes. *J. Qual. Technol.* **21**, 155–162 (1989)
57. Zhang, N.F.: A statistical control chart for stationary process data. *Technometrics* **40**, 24–38 (1998)
58. Capizzi, G., Masarotto, G.: An adaptive exponentially weighted moving average control chart. *Technometrics* **45**, 199–207 (2003)
59. Pascual, F.: EWMA charts for the weibull shape parameter. *J. Qual. Technol.* **42**, 400–416 (2010)
60. Zhang, L., Chen, G.: EWMA charts for monitoring the mean of censored Weibull lifetimes. *J. Qual. Technol.* **36**, 321–328 (2004)
61. Borror, C.M., Champ, C.W., Ridgon, S.E.: Poisson EWMA control charts. *J. Qual. Technol.* **30**, 352–361 (1998)
62. Gan, F.F.: Monitoring Poisson observations using modified exponentially weighted moving average control charts. *Commun. Stat. Simul. Comput.* **19**, 103–124 (1990)
63. Perry, M.B., Pignatiello Jr. J.J.: Estimation of the change point of the process fraction nonconforming in SPC applications. *Int. J. Reliab. Qual. Saf. Eng.* **12**, 95–110 (2005)
64. Sparks, R.S., Keighley, T., Muscatello, D.: Optimal exponentially weighted moving average (EWMA) plans for detecting seasonal epidemics when faced with non-homogeneous negative binomial counts. *J. Appl. Stat.* **38**, 2165–2181 (2011)
65. Gan, F.F.: EWMA control chart under linear drift. *J. Stat. Comput. Simul.* **38**, 181–200 (1991)
66. Gombay, E.: Sequential change-point detection and estimation. *Seq. Anal.* **22**, 203–222 (2003)
67. Hinkley, D.V.: Inference about the change-point in a sequence of random variables. *Biometrika* **57**, 1–17 (1970)
68. Grahamy, M.A., Human, S.W., Chakraborti, S.: A Phase I nonparametric Shewhart-type control chart based on median. *J. Appl. Stat.* **37**, 1795–1813 (2010)
69. Jones-Farmer, L.A., Jordan, V., Champ, C.W.: Distribution-free phase I control charts for subgroup location. *J. Qual. Technol.* **41**, 304–317 (2009)
70. Ansari, A.R., Bradley, R.A.: Rank-sum tests for dispersions. *Ann. Math. Stat.* **31**(4), 1174–1189 (1960)
71. Anderson, T.W.: On the distribution of the two-sample Cramer-von Mises criterion. *Ann. Math. Stat.* **33**(3), 1148–1159 (1962)
72. Anderson, T.W., Darling, D.A.: Asymptotic theory of certain ‘goodness of fit’ criteria based on stochastic processes. *Ann. Math. Stat.* **23**, 193–212 (1952)
73. Xiao, Y.: (2012). <https://cran.r-project.org/src/contrib/Archive/CvM2SL2Test/>
74. Xiao, Y., Gordon, A., Yakovlev, A.: A C++ program for the Cramer-von Mises two sample test. *J. Stat. Softw.* **17**(8) (2007)
75. Mood, A.: On the asymptotic efficiency of certain nonparametric two-sample tests. *Ann. Math. Stat.* **25**, 514–533 (1954)
76. Lepage, Y.: Combination of Wilcoxiens and Ansari–Bradley Statistics. *Biometrika* **58**, 213–217 (1971)
77. Lichman, M.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine (2013). <http://archive.ics.uci.edu/ml>



Partha Sarathi Mukherjee received his PhD in Statistics from University of Minnesota at Twin Cities, USA in 2011. He worked as a faculty at Boise State University, USA from 2012 to 2019. Recently, he joined Indian Statistical Institute, Kolkata, India as a faculty. He works on statistical image restoration, statistical process control, and application of statistical tools in various disciplines.

Reliability Models and Survival Analysis



Reliability Characteristics and Optimization of Warm Standby Systems

20

Suprasad V. Amari  and Hoang Pham

Contents

20.1	Introduction	385
20.2	Warm Standby Model and Related Works	387
20.2.1	Model Description and Assumptions.....	387
20.2.2	Related Works.....	387
20.3	Reliability Analysis	388
20.3.1	State Probabilities.....	388
20.3.2	System Reliability.....	390
20.3.3	Nonidentical Operational Failure Rates.....	391
20.3.4	Switch Failures.....	391
20.3.5	Reliability Improvement.....	392
20.4	Reliability Characteristics	393
20.4.1	Cumulative Distribution Function.....	393
20.4.2	Probability Density Function.....	393
20.4.3	Failure Rate Function.....	394
20.4.4	System Time to Failure Distribution.....	394
20.4.5	Moment-Generating and Characteristic Functions.....	394
20.4.6	Mean and Variance.....	395
20.4.7	Moments.....	395
20.4.8	Skewness and Kurtosis.....	395
20.4.9	Rényi and Shannon Entropies.....	396
20.4.10	Mean Residual Life Function.....	396
20.4.11	Mean Past Life or Mean Inactive Time Function.....	396
20.5	Optimal Reliability Design	397
20.5.1	Optimization Model Description.....	397
20.5.2	Optimization Model Formulation.....	397
20.5.3	Optimal Allocation of Fixed Number of Total Spares.....	398
20.5.4	Optimal Allocation with Different Resource Consumption Rates.....	399
20.5.5	Numerical Example.....	400
20.6	Conclusions	400
References	401

Abstract

This chapter presents reliability characteristics and optimal redundancy allocation of k -out-of- n warm standby systems consisting of identical components having exponential time to failure distributions. It is shown that the state probabilities of the warm standby system can be represented using the formulas that are applicable for active redundancy system. Subsequently, it is shown that all properties and computational procedures that are applicable for active redundancy are also applicable for the warm standby redundancy. The new results prove that the system reliability can be computed using robust and efficient computational algorithms with $O(n-k + 1)$ time complexity. Further, it is proved that the time-to-failure distribution of k -out-of- n warm standby system is beta-exponential. Using this property, closed-form expressions for various reliability characteristics and statistical measures of system failure time are presented. It has shown that the system reliability function is log-concave in n and this property is used to find efficient algorithms for determining optimal system configurations. Because active redundancy is a special case of the warm standby redundancy, indirectly this chapter also provides some new results for the active redundancy case as well.

Keywords

System reliability · Warm standby system · Log-concave property · k -out-of- n redundancy · Beta-exponential distribution · Regularized incomplete beta function · Optimization

20.1 Introduction

Standby redundancy is one of the primary techniques used to improve the system reliability to meet the requirements. Warm standby is a generic form of redundancy and it has been

S. V. Amari (✉)
BAE Systems, Merrimack, NH, USA

H. Pham
Department of Industrial and Systems Engineering, Rutgers University, Piscataway, NJ, USA

used in a wide range of applications and systems including power systems applications [1], space applications [2], and telecommunications systems [3]. Both hot and cold standby redundancies are special cases of warm standby redundancy [4]. In a cold standby system, inactive components have a zero failure rate, whereas in a hot standby system, both active and redundant (inactive) components have the same failure rates. Examples of a hot standby redundancy include a computing system with multiple servers that operate simultaneously and mirror the data in real time. In this case, both primary and redundant (secondary) servers have the same failure rate. In a warm standby redundancy, the inactive components have a failure rate that falls between cold and hot, inclusive. When switch delays and failures are not considered, the stochastic models for hot standby and active redundancy structures are equivalent [5]. Hence, the active redundancy structure can be considered as a special case of the warm standby structure. Therefore, the reliability characteristics presented in this chapter can also be applicable for the k -out-of- n active redundancy case.

Due to the generic nature of the warm standby redundancy, it is difficult to analyze the systems with these configurations as compared to the corresponding systems with active redundancy as well as cold and hot standby redundancies. Therefore, there are very few articles on reliability modeling for standby redundancy [6–8], nonrepairable warm standby systems in particular. This is because warm standby redundancy introduces complex, temporal, dynamic dependencies among the component lifetimes, and, therefore, the independence assumption of components in active redundancy is no longer valid for these systems [9]. For example, Yearout et al. [10] reviewed and categorized 156 papers specifically focusing research in standby redundancy. However, only 4 out of 156 papers were on non-repairable standby systems and none of them addresses the warm standby redundancy.

She and Pecht [4] derived a general closed-form reliability expression for k -out-of- n warm standby systems with s -identical components following exponential failure time distributions. Gnedenko et al. [11] analyzed the 1-out-of- n warm standby redundancy considering identical components with arbitrary failure time distributions applying the concept of equivalent age associated with cumulative exposure model. Li et al. [12] discussed a warm standby system with component lives following proportional hazard rate model. Morrison and David [13] proposed an efficient method based on counting process to analyze the system reliability of k -out-of- n cold standby redundancy configurations with identical components subject to arbitrary failure time distributions. This method is also described in [14]. Integrating the concepts used in [11, 13], Amari and Dill [9] developed an efficient computational method based on counting processes to evaluate system reliability of k -out-of- n warm standby redundancy with components following arbitrary failure time

distributions. This method was further extended to analyzing systems with components following specific failure time distributions including the Erlang [15], gamma [16], and Rayleigh [17] distributions. Eryilmaz [18] studied reliability of a k -out-of- n system having a single warm standby redundant component where the standby unit is used only after consumption of all active redundant components. In other words, the standby unit is kept into operation after the failure of the k -out-of- n active system by replacing the last failed unit, and then the system works until the failure of one of the k remaining components. Therefore, it can be considered as a nonstandard model that is applicable to a very specific scenario.

Using the reliability evaluation algorithm presented in [9], Amari and Dill [19] studied the constrained redundancy optimization problem for a series system consisting of multiple k -out-of- n warm standby subsystems. Levitin et al. [20] studied the 1-out-of- n :G cold standby systems subject to imperfect even backups and investigated the effects of backup system reliability in connection with other system parameters including data volume, frequency and duration of backups, replacement time and failure probability, and number of system elements. Wang et al. [21] presented a redundancy allocation problem for maximizing system reliability considering cold standby redundancy and degrading components.

Recently, the use of cold/warm standby systems has been expanded to several application areas. Tannous et al. [22] studied the reliability of warm standby systems subjected to imperfect fault coverage in the context of wireless sensor networks. For the identical component case, Amari et al. [23] provided closed-form expressions for reliability of warm standby systems subjected to imperfect coverage. Amari et al. [24] studied warm standby systems in the context of phased mission systems that are used in aerospace and defense applications. Levitin [25] studied the consecutive sliding window systems with warm standby components considering dynamic demands. When nonidentical cold/warm standby spares are used, the reliability and energy consumption of the system can depend on the sequence of standby units used in the operation [26]. Boddu et al. [27] considered optimal sequencing problem considering k -out-of- n :G cold standby systems. Levitin et al. [28] studied the optimal configuration of a series system consisting of 1-out-of- n warm standby subsystems performing operation and rescue functions. Further, Levitin et al. [29] studied heterogeneous warm standby systems and analyzed the effects of failure propagation on mission abort policy. They also [30] analyzed the dynamic checkpointing policy in heterogeneous real-time standby systems subjected to a constraint on allowed task completion time in performing the failure recovery. Jia et al. [31] studied the reliability modeling of standby systems considering multistate components subject to constant state transition rates.

However, as mentioned earlier, reliability characteristics of standby redundancy, in particular for the warm standby configurations, are less explored. For example, most of the research contributions on ageing properties are limited to active redundancy systems. Due to the complex nature of warm standby configurations, neither the closed-form expressions nor closure properties of mean residual life (MRL), probability density function (pdf), and hazard rate function/s/classes are studied for warm standby systems. In addition to studying these properties, it is important to derive closed-form expressions for pdf, hazard (failure) rate, reliability, and MRL functions that can lead to robust and computationally efficient evaluation of these fundamental reliability characteristics for warm standby systems.

This chapter presents recent results on reliability characteristics of k -out-of- n warm standby systems with s -identical components following exponential lifetimes. This model was initially studied in [4]. Later Amari et al. [23, 32] have presented closed-form expressions and efficient computational methods for reliability characteristics of warm standby systems. These results provide new insights into the system behavior and aging properties that in turn lead to efficient algorithms for finding optimal system configurations [33]. This chapter summarizes these results and then provide a set of new formulas on various statistical properties of the system failure time distribution.

20.2 Warm Standby Model and Related Works

Consider a k -out-of- n warm standby system model studied in [4].

20.2.1 Model Description and Assumptions

1. The system consists of n identical components.
2. Initially all components are in good conditional.
3. There shall be a minimum of k operating components for a successful operation.
4. The system redundancy configuration is k -out-of- n :G warm standby.
5. As long as the system is working, only k components are in operation and the remaining components are kept in the warm standby (dormant) mode.
6. Each component has a constant operating-failure rate (λ_o) and a constant dormant-failure rate (λ_d). Usually, $\lambda_d \leq \lambda_o$, and in many practical cases, λ_d is much smaller than λ_o . However, the analysis provided in this chapter is also applicable even when $\lambda_d > \lambda_o$.
7. There are no simultaneous failure of components.
8. Components are nonrepairable (no repair).

9. No sensing and switching delays (instantaneous).
10. Perfect switches (failure free). Note: Imperfect switches are considered in Sect. 20.3.4 by relaxing this assumption.

20.2.2 Related Works

Considering failure time of each component follows the exponential distribution with rate parameter λ_o in the operational state and rate parameter λ_d in the standby state, She and Pecht [4] provided a closed-form expression for system reliability at time t for the warm standby model described in Sect. 20.2.1:

$$\begin{aligned}
 R(t) &\equiv R(t; k, n, \lambda_o, \lambda_d) \\
 &= \frac{1}{(n-k)! \lambda_d^{n-k}} \sum_{i=0}^{n-k} (-1)^i \binom{n-k}{i} \\
 &\quad \times \left[\prod_{j=0, j \neq i}^{n-k} (k\lambda_o + j\lambda_d) \right] \exp \{-(k\lambda_o + i\lambda_d) t\}
 \end{aligned} \tag{20.1}$$

In addition to standby systems, Eq. (20.1) can also be used to evaluate the reliability of system with active redundancy. However, Eq. (20.1) cannot be used directly to evaluate the reliability of a cold standby system, but it should be evaluated at the limit $\lambda_d \rightarrow 0$. At this limit, the reliability expression becomes the survival function (Sf) of the gamma distribution. Hence

$$\begin{aligned}
 R(t) &\equiv \lim_{\lambda_d \rightarrow 0} R(t; k, n, \lambda_o, \lambda_d) \\
 &= \exp \{-k\lambda_o t\} \left[\sum_{i=0}^{n-k} \frac{(k\lambda_o t)^i}{i!} \right] \\
 &= \text{poif}(k\lambda_o t, n-k) \\
 &= \text{gamfc}(k\lambda_o t, n-k+1)
 \end{aligned} \tag{20.2}$$

where $\text{poif}(\cdot)$ is the cdf of Poisson distribution and $\text{gamfc}(\cdot)$ is the Sf of Erlang (gamma with integer shape parameter) distribution and these are available in all standard statistical packages.

Although Eq. (20.1) provides a theoretically exact expression for system reliability, it is unstable for large values of $(n-k)$ and/or small values of $\frac{\lambda_d}{\lambda_o}$, and therefore, it cannot be used directly for these cases. This is because Eq. (20.1) is prone to numerical roundoff errors due to the presence of alternative positive and negative terms [23]. The results of experiments presented in [23] indicate that, in some cases, the computed reliability values are not even within the valid

range for the reliability (probability), i.e., [0,1]. Therefore, the system reliability should be expressed in the forms that can lead to computationally efficient and robust numerical evaluation. A detailed discussion on numerical instability of Eq. (20.1) and the efficiency and robustness of the new results are available in [23, Section IV].

Amari and Pham [32] studied a special case of the warm standby system where $k_d \equiv k\lambda_o/\lambda_d$ is an integer. For this special case, it was shown that the Markov state transition diagram of this model is the same as that of k_d -out-of- $(n-k+k_d)$ active redundancy configuration where hazard (failure) rate of each component is λ_d . Hence, it was shown that the warm standby configuration behaves like an active redundancy configuration with modified parameters. For example, consider a 2-out-of-4 warm standby system where $\lambda_o = 3$ and $\lambda_d = 0.5$. Hence, $k_d \equiv 2 \cdot (3/0.5) = 12$ and $n_d = n - k + k_d = 5 - 3 + 12 = 14$. Hence, from Amari and Pham [32], it behaves like a 12-out-of-14 active redundancy system where the component failure rate is $\lambda_d = 0.5$. Hence, the system reliability function, state probabilities, system hazard (failure) rate function, pdf, MRL function, and failure time moments of the 2-out-of-4 warm standby system (with $\lambda_o = 3$ and $\lambda_d = 0.5$) are equivalent to the corresponding functions for the 12-out-of-14 active redundancy configuration with modified parameters (i.e., failure rate equal to $\lambda = 0.3$).

Further, Amari et al. [23] considered the general case of warm standby system, where $k_d \equiv k\lambda_o/\lambda_d$ is not restricted to an integer, and proved the following results:

- All properties and computational procedures that are available for active redundancy case are also applicable for the warm standby configurations.
- Reliability of both active redundancy and warm standby systems can be expressed and evaluated using beta distribution.
- The system reliability improvement with an additional redundant component follows negative binomial (Pólya) distribution.

In addition, Levitin et al. [23] expressed the system reliability in several forms that can allow and enable robust and efficient evaluation of system reliability, state probabilities, and reliability improvement factors with additional spares. Similarly, closed expressions were provided for higher order moments of the system failure time. Further, it was shown that the hazard rate of the warm system is a monotonically increasing function. Similarly, Levitin et al. [23] provided several closed-form expressions for the MRL function, and some of these forms also provide new and compact closed-form expressions for the MRL function of the k -out-of- n system with active redundancy [8]. In addition, it was shown that the system failure time follows beta-exponential distribution [34]. Using these results, some additional closed-

form expressions are derived in this chapter. These new closed-form expressions derived include moment-generating function, characteristic function, variance, cumulants, moments, skewness, kurtosis, Rényi entropy, and Shannon entropy. The closed-form expression in [23] also allowed proving several log-concave properties and aging properties that are applicable for both active and standby redundancies.

20.3 Reliability Analysis

The system failure time can be expressed as a sum of exponentially distributed random variables as shown in the Markov state transition diagram in Fig. 20.1, where state- i corresponds to the system state with i failures (including active and dormant failures), state 0 is the initial state and the state $(n - k + 1)$ is a failed state.

20.3.1 State Probabilities

The Markov state transition diagram in Fig. 20.1 indicates that the system failure time (T) can be expressed as the sum of exponential random variables

$$T = X_0 + X_1 + \dots + X_{n-k} \tag{20.3}$$

where X_i is the time spent in state i ; $i = 0, \dots, (n - k)$ and it follows exponential distribution with rate parameter $\lambda_i = k\lambda_o + (n - k - i)\lambda_d$. Further, the Laplace transform of probability of state- i , i.e., $P_i(t)$, can be expressed as

$$\mathcal{L}\{P_i(t)\} = \frac{B_i}{\prod_{j=0}^i (s + \lambda_j)}; \quad i = 0, \dots, n - k$$

$$B_i \equiv \prod_{j=0}^{i-1} \lambda_j; \quad i = 1, \dots, n - k \tag{20.4}$$

where $\lambda_j = k\lambda_o + (n - k - j)\lambda_d$ and $B_0 = 1$. Hence, using the closed-form expressions for inverse Laplace transforms in [35], $P_i(t)$ can be expressed as

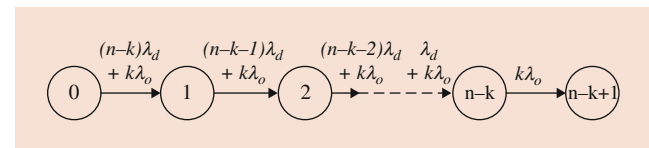


Fig. 20.1 Markov state transition diagram for k -out-of- n warm standby system [23, 32]

$$P_i(t) = B_i \sum_{j=0}^i A_{ij} \exp(-\lambda_j t); \quad i = 0, \dots, n - k \tag{20.5}$$

$$= \frac{\Gamma(n_d + 1)}{\Gamma(n_d - i + 1)} \exp(-\lambda_i t) \sum_{j=0}^i \frac{[-\exp\{-\lambda_d t\}]^{i-j}}{j!(i-j)!} \tag{20.8}$$

$$A_{ij} \equiv \prod_{\substack{m=0 \\ m \neq j}}^i \frac{1}{\lambda_m - \lambda_j}; \quad j = 0, \dots, i \tag{20.5}$$

Now Eq. (20.8) can be simplified using the following identity for the binomial expansion:

Further,

$$(1 - x)^i = \sum_{j=0}^i \frac{i!}{j!(i-j)!} (-x)^{i-j} \tag{20.9}$$

$$A_{ij} = \left[\prod_{m=0}^{j-1} \frac{1}{\lambda_m - \lambda_j} \right] \left[\prod_{m=j+1}^i \frac{1}{\lambda_m - \lambda_j} \right]$$

$$= \left[\frac{1}{j! \lambda_d^j} \right] \left[\frac{(-1)^{i-j}}{(i-j)! \lambda_d^{i-j}} \right]$$

$$= \frac{(-1)^{i-j}}{j!(i-j)! \lambda_d^i} \tag{20.6}$$

Hence

$$P_i(t) = \frac{\Gamma(n_d + 1)}{i! \Gamma(n_d - i + 1)} \exp\{-\lambda_i t\} [1 - \exp\{-\lambda_d t\}]^i \tag{20.10}$$

Similarly, we can simplify $B_i, i = 1, \dots, n - k$:

Further

$$\lambda_i = k\lambda_o + (n - k - i)\lambda_d = k_d\lambda_d + (n - k - i)\lambda_d = (n_d - i)\lambda_d \tag{20.11}$$

$$B_i = \prod_{j=0}^{i-1} \lambda_j$$

$$= \prod_{j=0}^{i-1} [k\lambda_o + (n - k - j)\lambda_d]$$

$$= \prod_{j=0}^{i-1} [k_d\lambda_d + (n - k - j)\lambda_d]$$

$$= \lambda_d^i \prod_{j=0}^{i-1} [(n - k + k_d) - j]$$

$$= \lambda_d^i \prod_{j=0}^{i-1} (n_d - j)$$

$$= \lambda_d^i \frac{\Gamma(n_d + 1)}{\Gamma(n_d - i + 1)} \tag{20.7}$$

Therefore

$$P_i(t) = \frac{\Gamma(n_d + 1)}{i! \Gamma(n_d - i + 1)} (\exp\{-\lambda_d t\})^{n_d - i} (1 - \exp\{-\lambda_d t\})^i$$

$$= \binom{n_d}{i} (p_d)^{n_d - i} (1 - p_d)^i \tag{20.12}$$

Note that n_d in Eq. (20.7) is not necessarily an integer. If $k_d \equiv k\lambda_o/\lambda_d$ is a real number, then $n_d \equiv n - k + k_d$ also becomes a real number. Hence, the gamma function in Eq. (20.7) can be replaced with a factorial function only when n_d is an integer. Substituting A_{ij} and B_i in Eqs. (20.6) and (20.7) into Eq. (20.5), we have

Note that the n_d in Eq. (20.12) can be a real number. Specifically, if $k_d \equiv k\lambda_o/\lambda_d$ is a real number, then $n_d \equiv n - k + k_d$ is also a real number. Therefore, the binomial coefficient in Eq. (20.12) is expressed using a real valued parameter n_d [36]. Further, note that Eq. (20.12) is also applicable even when $\lambda_d > \lambda_o$. For the active redundancy (hot standby), $\lambda_d = \lambda_o$. Further, $k_d = k, n_d = n, \Gamma(n_d + 1) = n!$, and $\Gamma(n_d - i + 1) = (n - i)!$. Hence, for the active redundancy, Eq. (20.12) reduces to the probability density function (pmf) of the binomial distribution [6, 7]. In addition, as shown in [32], (20.12) is equal to the pmf of the binomial distribution when $k_d = k\lambda_o/\lambda_d$ is an integer. Similarly, it is straightforward to prove that when $\lambda_d \rightarrow 0$ (for cold standby), Eq. (20.12) approaches the cdf of the Poisson distribution in (20.2). Similarly, for the cold standby system, $P_i(t)$ is equal to the pmf of the Poisson distribution. Hence, the state probability distribution in (20.12) includes binomial distribution as the special case and Poisson distribution as the limiting distribution. Hence, the new form of distribution for system state probabilities in (20.12) can be called a generalized truncated binomial distribution with a real valued n .

$$P_i(t) = \frac{\Gamma(n_d + 1)}{\Gamma(n_d - i + 1)} \sum_{j=0}^i \frac{(-1)^{i-j}}{j!(i-j)!} \exp(-\lambda_j t)$$

$$= \frac{\Gamma(n_d + 1)}{\Gamma(n_d - i + 1)} \exp(-\lambda_i t) \sum_{j=0}^i \frac{(-1)^{i-j}}{j!(i-j)!} \exp\{-(i-j)\lambda_d t\}$$

20.3.2 System Reliability

The system reliability can be evaluated as the sum of the probabilities of all good states. Hence, the reliability of the system is

$$\begin{aligned} R(t) &\equiv \sum_{i=0}^{n-k} \Pr\{\text{system is in state } i\} = \sum_{i=0}^{n-k} P_i(t) \\ &= \sum_{i=0}^{n-k} \frac{\Gamma(n_d + 1)}{i! \Gamma(n_d - i + 1)} (p_d)^{n_d - i} (1 - p_d)^i \\ &= \sum_{i=0}^{n-k} \binom{n_d}{i} (p_d)^{n_d - i} (1 - p_d)^i \end{aligned} \quad (20.13)$$

where $p_d \equiv p_d(t) = \exp(-\lambda_d t)$. Note that Eq. (20.13) is applicable for both real and integer values of $k_d \equiv k\lambda_o/\lambda_d$. In other words, $n_d \equiv n - k + k_d$ in Eq. (20.13) can be a real number and the binomial coefficient is defined for both real and integer valued arguments [36]. It is interesting to note that the reliability expression for warm standby system is similar in form to that of the active redundancy case. Therefore, the warm standby redundancy configuration can be viewed as an active redundancy configuration with modified parameters. These results may, at the first glance, seem to be counterintuitive or too good to be true. However, in addition to the detailed proofs provided in the previous section, the equivalence of results in Eqs. (20.1) and (20.13) can easily be cross-verified using numerical examples. In addition, the equivalence of these two expressions in (20.1) and (20.13) can easily be proved, because these are the two different forms of the same regularized incomplete beta function presented in Sections 26.5.6 and 26.5.7 of Abramowitz and Stegun [37], respectively. Hence, Eq. (20.1) is equivalent to Eq. (20.13) that was derived in She and Pecht [4]. However, the system reliability expressed using Eq. (20.13) provides more insight into the system reliability characteristics and leads to robust and computationally efficient numerical evaluation.

Note that $P_0 = p_d^{n_d}$ and $P_i = P_{i-1} \cdot \frac{n_d - i + 1}{i} \cdot \frac{1 - p_d}{p_d}$ for $i = 1, \dots, n - k$. Hence, the system reliability can be evaluated in $O(n-k+1)$ time complexity algorithms [6]. The algorithm evaluating the system reliability is described in Algorithm-A.

Algorithm-A: Reliability Evaluation Using State Probabilities

Inputs: $k, n, \lambda_o, \lambda_d, t$

$n_d = n - k + k\lambda_o/\lambda_d$; $p_d = \exp\{-\lambda_d t\}$

$x = p_d^{n_d}$; $R = x$; $y = (1 - p_d)/p_d$

for $i = 1$ to $(n - k)$ do

$$x = x \cdot y \cdot \frac{n_d - i + 1}{i}; \quad R = R + x$$

done

At the end of the algorithm, R contains the system reliability. At each iteration of i , x is equal to the probability of state i , $P_i(t)$. Hence, the system state probabilities ($P_i(t)$) can also be evaluated in $O(n-k+1)$ computational time. Further, for active redundancy case (when k_d is an integer), the system reliability can be evaluated with $O(\min\{k, n - k + 1\})$ computational time algorithms [6, 32].

Example 1 2-out-of-4 warm standby system where $\lambda_o = 0.0025$, $\lambda_d = 0.0004$, and $t = 100$. Hence, $k_d = 12.5$, $n_d = 14.5$, $(n-k) = 2$, $p_d = 0.960789$, and $\frac{1-p_d}{p_d} = 0.040811$. Further, $P_0 = p_d^{n_d} = (0.960789)^{14.5} = 0.5599$ and $P_i = P_{i-1} \cdot \frac{13.5-i}{i} \cdot (0.040811)$ for $i = 1, 2$. Hence, $P_1 = 0.33132$ and $P_2 = 0.091271$. Finally, the system reliability is $R = P_0 + P_1 + P_2 = 0.98249$.

Further, the system reliability (or unreliability) can be expressed in terms of either Sf or cdf of the beta distribution with parameters a and b . When b is an integer, the cdf of beta distribution, $F_B(x; a, b) \equiv I_x(a, b)$, can be expressed as [37]

$$I_x(a, b) = \sum_{i=0}^{b-1} \frac{\Gamma(a+b)}{\Gamma(i+1)\Gamma(a+b-i)} x^{a+b-1-i} (1-x)^i \quad (20.14)$$

By substituting $a = k_d$, $b = n_d - k_d + 1$, and $x = p_d$, we can express the system reliability in (20.13) using $I_x(a, b)$. Hence

$$R(t) \equiv I_{p_d}(k_d, n_d - k_d + 1) = F_B(p_d; k_d, n_d - k_d + 1) \quad (20.15)$$

where F_B is the cdf of the beta distribution and is available in all standard statistical packages. Further, there are several well-established numerical methods for computing the regularized incomplete beta function $I_x(a, b)$ [38], and this function is available in all standard mathematical software libraries including MATLAB[®], Mathematica[®], Mathcad[®], and Microsoft[®] Excel. In fact, reliability evaluation using (20.15) is preferred for large values of $(n - k)$. Because $I_x(a, b) = 1 - I_{(1-x)}(b, a)$ [37], the system reliability can also be expressed as the Sf of the beta distribution:

$$\begin{aligned} R(t) &\equiv 1 - I_{1-p_d}(n_d - k_d + 1, k_d) \\ &= 1 - F_B(1 - p_d; n_d - k_d + 1, k_d) \\ &= \bar{F}_B(1 - p_d; n_d - k_d + 1, k_d) \end{aligned} \quad (20.16)$$

where \bar{F}_B is the Sf of the beta distribution.

Example 2 Same as *Example 1*, except $R(t)$ is calculated from (20.15) and (20.16) using the MATLAB[®] function `betainc`. From (20.15), the reliability of the system can be calculated as $R(t) = I_{0.960789}(12.5, 3) = \text{betainc}(0.960789, 12.5, 3) = 0.98249$. Similarly, from (20.16), $R(t) = 1 - I_{0.039211}(3, 12.5) = \text{betainc}(0.039211, 3, 12.5, 'upper') = 0.98249$.

Note that $I_x(a, b)$ itself can be expressed in several forms [37]. Therefore, the reliability of warm standby system can be expressed in several forms. Some of these forms are used to derive other reliability characteristics.

20.3.3 Nonidentical Operational Failure Rates

Although the analysis presented in this chapter is based on the assumption that all units are identical, it can also be applied to the case where the active components experience nonidentical operational stresses. Specifically, the same reliability formula can also be applied to the situation where failure rates of k operational units are distinct (or not necessarily identical). In other words, even though the failure rate in the standby mode is the same for every unit (spare), the operational failure rate of a component can depend on its location or configuration of its operation. However, as long as the system is functioning, there will be the same number of operational units (k units) and the sum of their failure rates will be the same throughout the mission. Let $\lambda_{o,j}$ be the operational failure rate of a component located at position j . Because there are k operational units, the sum and average of their failure rates can be computed as $\lambda_{o,sum}$ and $\lambda_{o,avg}$:

$$\begin{aligned}\lambda_{o,sum} &\equiv \sum_{j=1}^k \lambda_{o,j} \\ \lambda_{o,avg} &\equiv \frac{1}{k} \sum_{j=1}^k \lambda_{o,j}\end{aligned}\quad (20.17)$$

These failure rates, i.e., $\lambda_{o,sum}$ and $\lambda_{o,avg}$, are the same for any combination of component failures (i.e., system state). Hence, the state transition diagram in Fig. 20.1 is still applicable for modeling this situation. However, the operational failure rate λ_o in Fig. 20.1 should be replaced with $\lambda_{o,avg}$. Hence, as long as the average of operational rates is used

in the reliability formula, the same analysis can be used even when the operational units experience distinct location-dependent failure rates. Hence, Algorithm-A can also be used for evaluating the system reliability with nonidentical operational failure rates.

20.3.4 Switch Failures

In a standby redundancy configuration, upon a failure of an operational component, the failed component is replaced by a redundant component when available. If the failure occurs in a dormant component, it will be removed. Therefore, a warm standby redundancy configuration requires a switching mechanism to sense the presence of a failed component and to activate a redundant component, if one is available. However, the switch itself can fail on demand, i.e., it may fail to perform its operation upon a failure of a component, which in turn leads to system failure. When the system reliability is expressed as a sum of probabilities of good states, it is straightforward to include the effects of switch failures in the reliability expression [9]. Let p_{sw} be the switch reliability on demand. If there are exactly i failures in the system, the switch needs to operate successfully at all these i requests (corresponding to i failures). Hence, the switch reliability for i requests is p_{sw}^i . Therefore, the reliability of the system with switch failures on demand is

$$R(t) \equiv \sum_{i=0}^{n-k} p_{sw}^i P_i(t) = \sum_{i=0}^{n-k} \binom{n_d}{i} (p_d)^{n_d-i} [p_{sw} (1 - p_d)]^i \quad (20.18)$$

The derivation of Eq. (20.18) is based on the assumption that the switch is needed at each instance of every component failure irrespective of its standby/operational status. Equation (20.18) has important applications in safety critical systems where all failed components must be detected, located, and permanently removed from the system to eliminate the failure propagation due to imperfect fault coverage [39]. However, if the switches are needed only for replacing the failed components in operation with the standby ones, but not the removal of the failed components in the dormant state, then the following equation shall be used (the proof is straightforward and the results can be verified using Markov models):

$$R(t) \equiv \sum_{i=0}^{n-k} \frac{\binom{n_d+k_d[1-p_{sw}]}{i}}{\binom{n_d}{i}} P_i(t) = \sum_{i=0}^{n-k} \binom{n_d+k_d[1-p_{sw}]}{i} (p_d)^{n_d-i} (1-p_d)^i \quad (20.19)$$

Similarly, it is straightforward to modify (20.13) to include the effects of various types of imperfect fault coverage models [39].

20.3.5 Reliability Improvement

It is a well-known fact that increasing the number of spares increases the system reliability. Specifically, when k is fixed, system reliability is an increasing function in n . Understanding the properties of reliability improvement (expressed in terms of n) is important in determining the optimal system configurations [40]. The following identity is used to derive the expression for reliability improvement [37]:

$$I_x(a, b + 1) = I_x(a, b) + \frac{\Gamma(a + b)}{\Gamma(a)\Gamma(b + 1)} x^a (1 - x)^b \quad (20.20)$$

Therefore

$$\begin{aligned} \nabla R(n; k) &\equiv \Delta R(n - 1; k) \equiv R(n; k) - R(n - 1; k) \\ &= \frac{\Gamma(n_d)}{\Gamma(k_d)\Gamma(n_d - k_d + 1)} (p_d)^{k_d} (1 - p_d)^{n_d - k_d} \\ &= \binom{n_d - 1}{k_d - 1} (p_d)^{k_d} (1 - p_d)^{n_d - k_d} \quad (20.21) \end{aligned}$$

It can easily be verified that (20.21) reduces to the reliability improvement equation available in [7, 41] for the active redundancy when $k_d = k$, $n_d = n$, and $p_d = p$. Note that Eq. (20.21) can also be derived using the pivotal decomposition because the warm standby model can be expressed in terms of an active redundancy model with modified parameters. Further, define

$$\begin{aligned} \delta_i &\equiv \nabla R(k + i; k) = \binom{k_d + i - 1}{i} (p_d)^{k_d} (1 - p_d)^i; \\ \text{for } i &= 0, 1, \dots \quad (20.22) \end{aligned}$$

where δ_i is the improvement in system reliability when the number of spares is increased from $(i - 1)$ to i . Further, $R(k, n) = 0$ for $n < k$, and $\delta_0 = R(k, k)$. Hence, the system reliability is the sum of δ_i 's:

$$R(k, n) = \sum_{i=0}^{n-k} \delta_i \quad (20.23)$$

Note that $\delta_0 = p_d^{k_d}$ and $\delta_i = \delta_{i-1} \cdot \frac{k_d + i - 1}{i} \cdot (1 - p_d)$ for $i = 1, \dots, n - k$. Hence, using (20.23), the system reliability can be computed in $O(n-k+1)$ computational time [6]. Refer to Algorithm-B for the detailed steps for evaluating the system reliability.

Algorithm-B: Reliability Evaluation Using Reliability Improvement Factors

Inputs: $k, n, \lambda_o, \lambda_d, t$

$$k_d = k\lambda_o/\lambda_d; \quad p_d = \exp\{-\lambda_d t\}$$

$$x = p_d^{k_d}; \quad R = x; \quad y = 1 - p_d$$

for $i = 1$ to $(n - k)$ do

$$x = x \cdot y \cdot \frac{k_d + i - 1}{i}; \quad R = R + x$$

done

At the end of the algorithm, R contains the system reliability. At each iteration of i , x is equal to the reliability improvement factor corresponding to $\delta_i(t)$. Hence, the reliability improvement factors can also be computed with $O(n-k+1)$ computational time algorithms.

Example 3 Same as *Example 1*, except $R(t)$ is calculated from (20.23). Here, $k_d = 12.5$, $(n-k) = 2$, $p_d = 0.960789$, and $(1 - p_d) = 0.039211$. Further, $\delta_0 = p_d^{k_d} = (0.96079)^{12.5} = 0.60653$ and $\delta_i = \delta_{i-1} \cdot \frac{11.5+i}{i} \cdot (0.039211)$ for $i = 1, 2$. Hence, $\delta_1 = 0.29728$ and $\delta_2 = 0.078682$. Finally, the system reliability is $R = \delta_0 + \delta_1 + \delta_2 = 0.98249$.

It is interesting to note that δ_i is nothing but the probability of state- i of the Markov chain shown in Fig. 20.2, where state-0 is the initial state and state- $(n - k + 1)$ is the final state. Note that the state transition diagrams in Figs. 20.1 and 20.2 are similar, except the occurrence of states are reversed. The time to reach the absorbing state in Fig. 20.1 is equivalent to the corresponding time in Fig. 20.2, because they both represent the sum of the same set of exponential random variables. When $n = k$, the state transition diagrams in Figs. 20.1 and 20.2 are equivalent. Hence, the probability of state-0 in Fig. 20.2 is equal to δ_0 . For $n = k + 1$, it should satisfy the following equation: $P_0(n = k + 1) + P_1(n = k + 1) = R(k, n = k + 1) = \delta_0 + \delta_1$. Hence, from Eq. (20.23), it can be shown that the probability of state-1 in Fig. 20.2 is equal to δ_1 . Using the similar arguments, it can be proved that the probability of state- i in Fig. 20.2 is equal to δ_i for any non-absorbing state- i .

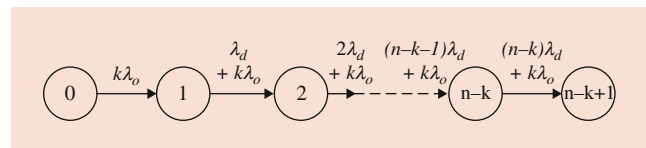


Fig. 20.2 Markov state transition diagram for k -out-of- n warm standby system

Note that δ_i in (20.22) is the pmf of the negative binomial (Pólya) distribution [42]. Therefore, the system reliability in (20.23) can be expressed as the cdf of the negative binomial distribution:

$$\delta_i = f_{NB}(i; p_d, k_d) \quad \text{and} \quad R(k, n) = \sum_{i=0}^{n-k} \delta_i = F_{NB}(n - k; p_d, k_d) \tag{20.24}$$

where $f_{NB}(x; p, r)$ and $F_{NB}(x; p, r)$ are the pmf and cdf of negative binomial distribution.

Although the negative binomial distribution is also defined for the positive real values of r , the applications for this case are rare. Hence, the warm standby system provides a new application for this distribution with a real valued parameter r . Note that the negative binomial is a unimodal distribution. Hence, δ_i attains its maximum value at the mode of the NB distribution. Define

$$m \equiv \max \left\{ 0, \left\lfloor \frac{(k_d - 1)(1 - p_d)}{p_d} \right\rfloor \right\} \tag{20.25}$$

Thus, the δ_i at first increases and then decreases, reaching its maximum value at $i = m$. If $\frac{(k-1)(1-p)}{p}$ is an integer, then $\delta_m = \delta_{m-1}$. The $\nabla R(n; k)$ in (20.21) and δ_i in (20.22) are equivalent, except that i represents the number of spares and $n \equiv k + i$ represents the total number of components. Therefore, $\nabla R(n; k)$ first increases and then decreases, reaching its maximum value at $n = n_{mode} = k + m$. Hence

$$\begin{aligned} n_{mode} &\equiv \max \left\{ k, \left\lfloor \frac{(k_d - 1)(1 - p_d)}{p_d} + k \right\rfloor \right\} \\ &= \max \left\{ k, \left\lfloor \frac{k_d - 1}{p_d} + k - k_d + 1 \right\rfloor \right\} \end{aligned} \tag{20.26}$$

It is interesting to note that $P_i = \delta_i$ for the cold standby system. This can be verified by comparing the individual terms in the cdf of the Poisson distribution. In addition, this equivalence can also be verified by comparing the state transition diagrams in Figs. 20.1 and 20.2 by setting $\lambda_d = 0$. Specifically, it can be easily observed that these two diagrams are equivalent when $\lambda_d = 0$. Hence, $P_i = \delta_i$. Further, using the properties of Poisson distribution, it can be shown that δ_i first increases and then decreases, reaching its maximum value at $i = m_s \equiv \max\{0, \lfloor k\lambda_o t \rfloor\}$.

Example 4 Case 1: Same as *Example 1*, except n is not specified. Here, $k = 2$, $k_d = 12.5$, and $p_d = 0.960789$. Therefore, from (20.26), $n_{mode} = 2$. Hence, $\nabla R(n; k) \equiv \nabla R(n)$ reaches its maximum value at $n = k = 2$, and it always decreases with increase in n . Case 2: Same as Case 1, except $t = 1000$. Hence, $p_d = 0.67032$ and $n_{mode} = 7$.

Therefore, $\nabla R(n; k) \equiv \nabla R(n)$ first increases for $n \leq 7$ and then decreases for $n > 7$, reaching its maximum value at $n = 7$.

It is well known that the pmf of the NB distribution is a log-concave function [43]. Hence, δ_{i+1}/δ_i is a decreasing function in i . In other words, $\delta_i^2 \geq \delta_{i-1}\delta_{i+1}$. Hence, from [43, Theorem 3.1], it can be shown that the cdf of NB distribution is also log-concave. Therefore, from (20.24), $R(n)$ is log-concave in n . In other words, $\frac{R(n+1)}{R(n)}$ is a decreasing function in n . Hence, the reliability improvement percentage with an additional component decreases with the number of redundant components. This log-concave property allows us to apply efficient methods to find optimal redundancy levels [44]. Therefore, the optimal spares allocation procedure proposed in [44] for the series system with 1-out-of- n cold standby subsystems can be extended for the series systems with k -out-of- n warm standby subsystems with exponentially distributed component lifetimes. These details are provided in Sect. 20.5.

20.4 Reliability Characteristics

20.4.1 Cumulative Distribution Function

The cdf, $F(t)$, is equivalent to the system unreliability, $Q(t)$:

$$\begin{aligned} F(t) &= Q(t; k, n) \equiv 1 - R(t; k, n) \\ &= 1 - I_{p_d}(k_d, n_d - k_d + 1) \\ &= I_{1-p_d}(n_d - k_d + 1, k_d) \end{aligned} \tag{20.27}$$

20.4.2 Probability Density Function

The pdf, $f(t)$, can be found as a first-order derivative of $F(t)$ with respect to t :

$$f(t; k, n) \equiv \frac{dF(t)}{dt} = -\frac{dR(t; k, n)}{dt} \tag{20.28}$$

For nonrepairable systems, pdf also equals to system failure intensity (frequency) [45]. From the Markov state transition diagram in Fig. 20.1, it can be noted that the system has only one failed state and it is the state $(n - k + 1)$. The system reaches state $(n - k + 1)$ only from state $(n - k)$. Hence, the transition from state $(n - k)$ to state $(n - k + 1)$ leads to the system failure. The state transition frequency is equal to the product of probability of state $(n - k)$ and the transition rate from state $(n - k)$ to state $(n - k + 1)$. Therefore, the failure frequency is $k\lambda_o P_{n-k}(t)$. Hence, the pdf is

$$\begin{aligned}
 f(t; k, n) &\equiv k\lambda_o P_{n-k}(t) = k_d\lambda_d P_{n-k}(t) \\
 &= k_d\lambda_d \binom{n_d}{n_d - k_d} [p_d]^{k_d} [1 - p_d]^{n_d - k_d} \\
 &= \frac{\lambda_d}{B(n_d - k_d + 1, k_d)} [\exp\{-\lambda_d t\}]^{k_d} [1 - \exp\{-\lambda_d t\}]^{n_d - k_d}
 \end{aligned}
 \tag{20.29}$$

Note that p_d and $(1 - p_d)$ are the Sf and cdf of an exponentially distributed *r.v.* Hence, these are log-concave in t [46]. Further, the product of two log-concave functions is also a log-concave function. Therefore, $f(t; k, n)$ is log-concave in t . Hence, the system reliability, $R(t; k, n)$, and system unreliability, $Q(t; k, n)$, are also log-concave in t [46]. Because $f(t; k, n)$ is log-concave, it can be shown that the system time to failure distribution is in the IFR class [46].

20.4.3 Failure Rate Function

The system failure (hazard) rate function can be determined as $h(t) = f(t)/R(t)$. Hence

$$\begin{aligned}
 h(t) &\equiv h(t; k, n) = \frac{f(t; k, n)}{R(t; k, n)} \\
 &= \frac{k_d\lambda_d \binom{n_d}{n_d - k_d} [p_d]^{k_d} [1 - p_d]^{n_d - k_d}}{\sum_{i=0}^{n-k} \binom{n_d}{i} [p_d]^{n_d - i} [1 - p_d]^i} \\
 &= \frac{k\lambda_o \binom{n_d}{n-k}}{\sum_{i=0}^{n-k} \binom{n_d}{i} [\theta(t)]^{n-k-i}}
 \end{aligned}
 \tag{20.30}$$

where $\theta(t) = \frac{p_d}{1-p_d} = \frac{\exp\{-\lambda_d t\}}{1-\exp\{-\lambda_d t\}}$. The $\theta(t)$ is a decreasing function in t . Hence, even without the knowledge of log-concave properties of pdf, it can be shown that $h(t)$ is strictly an increasing function in t when $k < n$ and is a constant (still IFR) when $k = n$. From (20.30), for any value of t , we have

$$0 = h(0) \leq h(t) \leq h(\infty) = k_d\lambda_d = k\lambda_o \tag{20.31}$$

20.4.4 System Time to Failure Distribution

From the Markov state transition diagram in Fig. 20.1, it can be shown that the system failure time, T , is the sum of independent exponential random variables. Therefore, from [42], it is a hypo-exponential distribution. Further, it can be shown that T follows phase-type (PH) distribution, because it can be viewed as the time to absorption of a finite-state Markov process. However, due to the generic nature of PH distributions, this property does not provide much insight of

the behavior of T . Amari et al. [23] have shown that T follows beta-exponential (BE) distribution proposed by Nadarajah and Kotz [34] as a generalization of beta distribution. The cdf of the BE distribution is

$$F_{BE}(x) = I_{1-\exp(-\lambda x)}(a, b) \tag{20.32}$$

for $x > 0, a > 0, b > 0$, and $\lambda > 0$. The pdf of BE *r.v.* is

$$f_{BE}(x) = \frac{\lambda}{B(a, b)} \exp(-b\lambda x) \{1 - \exp(-\lambda x)\}^{a-1} \tag{20.33}$$

Comparing the cdf expressions in (20.27) and (20.32), Amari et al. [23] have shown that these two expressions are identical when $a = n_d - k_d + 1$ and $b = k_d$. Hence, the failure time distribution of the k -out-of- n warm standby system, as well as its special case k -out-of- n active redundancy system, can be represented as a BE distribution with parameters $a = n_d - k_d + 1 = n - k + 1, b = k_d = k\lambda_o/\lambda_d$, and $\lambda = \lambda_d$. The same results can also be obtained by comparing the pdf expressions in (20.29) and (20.33). For the BE distribution, Nadarajah and Kotz [34] provided closed-form expressions for variance, characteristic function, moment-generating function, moments, cumulants, skewness, kurtosis, Rényi entropy, and Shannon entropy. Therefore, substituting $a = n_d - k_d + 1 = n - k + 1, b = k_d = k\lambda_o/\lambda_d$, and $\lambda = \lambda_d$, these functions can be represented directly in terms of the model parameters of the warm standby system. Note that $a = n - k + 1$ is an integer (even for the generic case). Hence, the corresponding expressions for warm standby systems can be simplified further.

20.4.5 Moment-Generating and Characteristic Functions

This subsection provides derivations for moment-generating function (mgf) and the characteristic function of random variable T having the pdf in Eq. (20.29). From Nadarajah and Kotz [34], the mgf is defined by $M(t) = E[\exp(tX)]$. Hence

$$M(t) = \frac{\lambda}{B(a, b)} \int_0^\infty \exp\{(t - b\lambda)x\} \{1 - \exp(-\lambda x)\}^{a-1} dx \tag{20.34}$$

By substituting $y = \exp(-\lambda x)$, the integral on the right-hand side reduces to

$$\int_0^1 y^{b-t/\lambda-1} (1 - y)^{a-1} dy = \frac{1}{\lambda} B(b - t/\lambda, a) \tag{20.35}$$

Hence, the moment-generating function is

$$\begin{aligned}
 M(t) &= \frac{B(b - t/\lambda, a)}{B(a, b)} \\
 &= \frac{B(k_d - t/\lambda_d, n_d - k_d + 1)}{B(n_d - k_d + 1, k_d)} \\
 &= \frac{B(k\lambda_d - t/\lambda_d, n - k + 1)}{B(n - k + 1, k\lambda_0/\lambda_d)} \tag{20.36}
 \end{aligned}$$

Note that T can be expressed as a sum of s -independent exponential $r.v$. Therefore, $M(t)$ can also be expressed as

$$M(t) = \prod_{j=0}^{n-k} \left[1 - \frac{t}{(k_d + j)\lambda_d} \right]^{-1} \tag{20.37}$$

Hence, the characteristic function of X defined by $\varphi(t) = E[\exp(itX)]$ takes the following form [34]:

$$\begin{aligned}
 \varphi(t) &= \frac{B(b - it/\lambda, a)}{B(a, b)} \\
 &= \frac{B(k_d - it/\lambda_d, n_d - k_d + 1)}{B(n_d - k_d + 1, k_d)} \\
 &= \frac{B(k\lambda_d - it/\lambda_d, n - k + 1)}{B(n - k + 1, k\lambda_0/\lambda_d)} \tag{20.38}
 \end{aligned}$$

Further, $\varphi(t)$ can also be expressed as

$$\varphi(t) = \prod_{j=0}^{n-k} \left[1 - \frac{it}{(k_d + j)\lambda_d} \right]^{-1} \tag{20.39}$$

20.4.6 Mean and Variance

The mean of T can be written as [34]

$$\begin{aligned}
 E[T] &= \{\psi(a + b) - \psi(b)\}/\lambda_d = \{\psi(n_d + 1) - \psi(k_d)\}/\lambda_d \\
 &= \sum_{i=0}^{n-k} \frac{1}{(k_d + i)\lambda_d} = \sum_{i=0}^{n-k} \frac{1}{k\lambda_0 + i\lambda_d} \tag{20.40}
 \end{aligned}$$

The variance of T can be written as

$$\begin{aligned}
 \text{Var}[T] &= \{\psi'(b) - \psi'(a + b)\}/\lambda_d^2 = \{\psi'(k_d) - \psi'(n_d + 1)\}/\lambda_d^2 \\
 &= \sum_{i=0}^{n-k} \frac{1}{[(k_d + i)\lambda_d]^2} = \sum_{i=0}^{n-k} \frac{1}{[k\lambda_0 + i\lambda_d]^2} \tag{20.41}
 \end{aligned}$$

20.4.7 Moments

As shown in Nadarajah and Kotz [34], it is immediate from Eq. (20.29) that the r^{th} moment of T can be written as

$$E[T^r] = \frac{(-1)^r}{\lambda^r B(a, b)} \frac{\partial^r B(a, 1 + p - a)}{\partial p^r} \Bigg|_{p=a+b-1} \tag{20.42}$$

Because $a \equiv n - k + 1$ is an integer, it can be simplified further for the warm standby system. The explicit expression for r^{th} moment of T can be written as [23]

$$\begin{aligned}
 E[T^r] &= \sum_{i=0}^{n-k} \binom{n}{i + k_d} \binom{i + k_d - 1}{i} (-1)^i \frac{r!}{(i + k_d)^r \lambda_d^r} \\
 &= \frac{r! \lambda_d}{B(k\lambda_0/\lambda_d, n - k + 1)} \sum_{i=0}^{n-k} \binom{n - k}{i} \frac{(-1)^i}{(k\lambda_0 + i\lambda_d)^{r+1}} \tag{20.43}
 \end{aligned}$$

Equation (20.43) has several applications in reliability field: (1) estimate parameters of the system using method of moments [34], (2) determine bounds and approximations for reliability of a complex system [47], and (3) dependability analysis considering reliability and performance measures simultaneously [42].

20.4.8 Skewness and Kurtosis

The skewness T can be written as

$$\begin{aligned}
 \text{Skewness}(T) &= \frac{\psi''(a + b) - \psi''(b)}{\{\psi'(b) - \psi'(a + b)\}^{3/2}} \\
 &= \frac{\psi''(n_d + 1) - \psi''(k_d)}{\{\psi'(k_d) - \psi'(n_d + 1)\}^{3/2}} \tag{20.44}
 \end{aligned}$$

The kurtosis of T can be written as

$$\begin{aligned}
 \text{Kurtosis}(T) &= \frac{3\{\psi'(b)\}^2 - 6\{\psi'(b)\psi'(a + b)\} + 3\{\psi'(a + b)\}^2 + \psi'''(b) - \psi'''(a + b)}{\{\psi'(b) - \psi'(a + b)\}^2} \\
 &= \frac{3\{\psi'(b)\}^2 - 6\{\psi'(k_d)\psi'(n_d + 1)\} + 3\{\psi'(n_d + 1)\}^2 + \psi'''(k_d) - \psi'''(n_d + 1)}{\{\psi'(k_d) - \psi'(n_d + 1)\}^2} \tag{20.45}
 \end{aligned}$$

20.4.9 Rényi and Shannon Entropies

An entropy of a random variable T is a measure of variation of the uncertainty [34]. Rényi entropy of order γ is defined by

$$H_\gamma(T) = \frac{1}{1-\gamma} \ln \left\{ \int f^\gamma(x) dx \right\} \quad (20.46)$$

where $\gamma \geq 0$ and $\gamma \neq 1$. For the BE distribution, it can be expressed as [34]

$$H_\gamma(T) = -\ln \lambda + \frac{1}{1-\gamma} \ln \left[\frac{B(\gamma a - \gamma + 1, \gamma b)}{B^\gamma(a, b)} \right] \quad (20.47)$$

Hence, for the warm standby system, it can be expressed as

$$\begin{aligned} H_\gamma(T) &= -\ln \lambda_d + \frac{1}{1-\gamma} \ln \left[\frac{B(\gamma(n_d - k_d) + 1, \gamma k_d)}{B^\gamma(n_d - k_d + 1, k_d)} \right] \\ &= -\ln \lambda_d + \frac{1}{1-\gamma} \ln \left[\frac{B(\gamma(n - k) + 1, \gamma k \lambda_o / \lambda_d)}{B^\gamma(n - k + 1, k \lambda_o / \lambda_d)} \right] \end{aligned} \quad (20.48)$$

Shannon entropy defined by $H(X) \equiv E[-\ln f(X)]$ is the particular case of (20.46) for $\gamma \rightarrow 1$. Limiting $\gamma \rightarrow 1$ in (20.48) and using L'Hospital's rule, the Shannon entropy for warm standby system can be derived:

$$\begin{aligned} H(T) &= E[-\ln f(T)] = -\ln \lambda_d + \ln B(a, b) + (a + b - 1)\psi(a + b) - (a - 1)\psi(a) - b\psi(b) \\ &= -\ln \lambda_d + \ln B(n_d - k_d + 1, k_d) + (n_d)\psi(n_d + 1) - (n_d - k_d)\psi(n_d - k_d + 1) - b\psi(k_d) \end{aligned} \quad (20.49)$$

20.4.10 Mean Residual Life Function

Mean residual life (MRL) is the expected remaining lifetime of the system given that it has survived until time t . Specifically, MRL is defined as $m(t) = E[T - t | T > t]$.

$$m(t) \equiv E[T - t | T > t] = \frac{\int_t^\infty R(x) dx}{R(t)} \quad (20.50)$$

The MRL function of k -out-of- n warm standby system can be represented in several forms. Some of these forms also provide new and compact expressions for the active redundancy model [8]:

$$\begin{aligned} m(t) &= \frac{\sum_{i=0}^{n-k} P_i(t) \mu_i}{\sum_{i=0}^{n-k} P_i(t)} \\ &= \frac{\sum_{i=0}^{n-k} \frac{R(t; k+i, n)}{(k_d+i)\lambda_d}}{R(t; k, n)} \\ &= \frac{\frac{1}{\lambda_d} \sum_{i=0}^{n-k} \binom{n_d}{i+k_d} \binom{i+k_d-1}{i} (-1)^i \frac{p_d^{i+k_d}}{(k_d+i)}}{\sum_{i=0}^{n-k} \binom{n_d}{i+k_d} \binom{i+k_d-1}{i} (-1)^i p_d^{i+k_d}} \end{aligned} \quad (20.51)$$

where μ_i is the MTTF of k -out-of- $(n - i)$ warm standby system: $\mu_i = \sum_{j=0}^{n-k-i} \frac{1}{k\lambda_o + j\lambda_d}$. The MRL expressions can be simplified by canceling the like terms in the numerator and the denominator:

$$m(t) = \frac{\sum_{i=0}^{n-k} \binom{n_d}{i} [\phi(t)]^i \mu_i}{\sum_{i=0}^{n-k} \binom{n_d}{i} [\phi(t)]^i}$$

$$\begin{aligned} &= \frac{\sum_{i=0}^{n-k} \binom{n_d}{i} [\theta(t)]^{k_d+i} \mu_{n-k-i}}{\sum_{i=0}^{n-k} \binom{n_d}{i} [\theta(t)]^{k_d+i}} \\ &= \frac{1}{\lambda_d} \frac{\sum_{i=0}^{n-k} (-1)^i \binom{n-k}{i} \frac{p_d^i}{(k_d+i)^2}}{\sum_{i=0}^{n-k} (-1)^i \binom{n-k}{i} \frac{p_d^i}{k_d+i}} \end{aligned} \quad (20.52)$$

where $\phi(t) = \frac{1-p_d}{p_d} = \frac{1-\exp\{-\lambda_d t\}}{\exp\{-\lambda_d t\}} = \exp\{\lambda_d t\} - 1$ and $\theta(t) = \frac{p_d}{1-p_d} = \frac{1}{\phi(t)}$. Because the system time to failure distribution is IFR, from [46], it can be shown that $m(t)$ is a decreasing function in t . Further, it can be shown that for any t :

$$\sum_{i=0}^{n-k} \frac{1}{k\lambda_o + i\lambda_d} = \mu = m(0) \geq m(t) \geq m(\infty) = \frac{1}{k_d\lambda_d} = \frac{1}{k\lambda_o} \quad (20.53)$$

20.4.11 Mean Past Life or Mean Inactive Time Function

The mean past life (MPL), also called mean inactive time or mean elapsed time, corresponds to the mean time elapsed since the failure of the system at time T until the observed time t given that $T \leq t$ [48, 49]. In this case, the random variable of interest is $\{t - T | T \leq t\}$. This conditional random variable is elapsed time since the failure of the system at T given that it has failed at or before t . The expectation of this random variable, denote by $\kappa(t)$, is known as the MPL. Specifically, MPL is defined as $\kappa(t) = E[t - T | T \leq t]$:

$$\kappa(t) \equiv E[t - T | T \leq t] = \frac{\int_0^t F(x) dx}{F(t)} \quad (20.54)$$

The MPL can be expressed in terms of MRL function, $m(t)$, and mean life function, $E[T]$:

$$\begin{aligned} \kappa(t) &= \frac{\int_0^t F(x) dx}{F(t)} = \frac{t - \int_0^t R(x) dx}{F(t)} \\ &= \frac{t - \int_0^\infty R(x) dx + \int_t^\infty R(x) dx}{F(t)} \\ &= \frac{t - E[T] + m(t)R(t)}{F(t)} \end{aligned} \quad (20.55)$$

Substituting $m(t)$ and $R(t)$ in (20.55), the MPL can be expressed as

$$\begin{aligned} \kappa(t) &= \frac{t - E[T] + \sum_{i=0}^{n-k} P_i(t) \mu_i}{1 - \sum_{i=0}^{n-k} P_i(t)} \\ &= \frac{t - \sum_{i=0}^{n-k} \frac{1}{(k_d+i)\lambda_d} + \sum_{i=0}^{n-k} \frac{R(t; k+i, n)}{(k_d+i)\lambda_d}}{1 - R(t; k, n)} \\ &= \frac{t - \sum_{i=0}^{n-k} \frac{1 - R(t; k+i, n)}{(k_d+i)\lambda_d}}{1 - R(t; k, n)} \\ &= \frac{t - \sum_{i=0}^{n-k} \frac{F(t; k+i, n)}{(k_d+i)\lambda_d}}{F(t; k, n)} \end{aligned} \quad (20.56)$$

where μ_i is the MTTF of k -out-of- $(n-i)$ warm standby system $\mu_i = \sum_{j=0}^{n-k-i} \frac{1}{k\lambda_o + j\lambda_d}$ and $F(t; k+i, n)$ is the unreliability of $(k+i)$ -out-of- n system where $F(t; k+i, n) = 1 - R(t; k+i, n)$.

20.5 Optimal Reliability Design

A primary objective of reliability engineering is designing highly reliable systems considering available resources and design constraints [1, 2]. One of the primary techniques in improving reliability is employing redundancy (spares). However, improvements in reliability must be made without violating system constraints [1]. When redundancy is used to improve reliability of a system consisting of multiple subsystems, it is important to determine the optimal redundancy levels for each subsystem. In general, the optimal spares allocation is a well-known NP-hard problem [1, 2]. Hence, there are no efficient solutions for the generic case of large-scale systems. This is because the computational time to solve this problem increases exponentially with the system size. Hence, all existing methods are applicable to either small-scale problems or to find approximate solutions. This is the case even if system configuration is series structure where all subsystems are connected logically in a series. However, an effective use of log-concave properties can

improve the efficiency of redundancy allocation problem [44]. Specifically, using log-concave properties, Amari [33] proposed an efficient linear-time algorithm to find optimal design configurations for nonrepairable series systems with a single system-level constraint. This section describes the method proposed in [33].

20.5.1 Optimization Model Description

- the system has M subsystems in series. Hence, the system is good only if all of its subsystems are good.
- The subsystems are s -independent.
- The subsystems use a k -out-of- n cold/warm standby or active redundancy. The subsystem- i requires at least k_i components for successful operation.
- The components within each subsystem are S -identical.
- The type of components used can vary with the subsystems.
- Each component has its own operational and dormant (standby) failure rates as well as resource consumption rates.
- The systems and its components are nonrepairable.

The objective is to maximize the system reliability through optimal allocation of spares for different subsystems without violating the system-level constraints such as total system cost/budget, weight, or volume (only one type of constraint is considered).

20.5.2 Optimization Model Formulation

The objective function is the system reliability and it can be expressed as

$$R(t; \mathbf{k}, \mathbf{n}) = \prod_{i=1}^M R(t; k_i, n_i) \quad (20.57)$$

where the reliability of each subsystem can be evaluated using Eqs. (20.13), (20.15), or (20.16).

The optimization method presented here is based on the log-concave properties of the warm standby systems discussed in Sect. 20.3.5. In other words, the optimization method is based on the property that $\frac{R(n+1)}{R(n)}$ is a decreasing function in n . To demonstrate the role of the log-concave properties, a 2-out-of- n warm standby system with varying number of components is considered where $(n-2)$ is the number of spares. Let $\lambda_o = 1$, $\lambda_d = 0.2$, and $t = 1$. The system reliability and its improvement factors at different number of spares are shown in Table 20.1. The results demonstrate that both $\frac{\delta_{i+1}}{\delta_i}$ and $\frac{R(n+1)}{R(n)}$ are decreasing functions.

Table 20.1 Log-concave properties of 2-out-of- n warm standby system

n	$i = n - k$	δ_i	$R(n)$	$\frac{\delta_{i+1}}{\delta_i}$	$\frac{R(n+1)}{R(n)}$
2	0	0.1353	0.1353	1.8127	2.8127
3	1	0.2453	0.3807	0.9970	1.6425
4	2	0.2446	0.6252	0.7251	1.2836
5	3	0.1773	0.8026	0.5891	1.1302
6	4	0.1045	0.9071	0.5076	1.0585
7	5	0.0530	0.9601	0.4532	1.0250

Table 20.2 Log-concave properties of 2-out-of- n cold standby system

n	$i = n - k$	δ_i	$R(n)$	$\frac{\delta_{i+1}}{\delta_i}$	$\frac{R(n+1)}{R(n)}$
2	0	0.1353	0.1353	2.0000	3.0000
3	1	0.2707	0.4060	1.0000	1.6667
4	2	0.2707	0.6767	0.6667	1.2667
5	3	0.1804	0.8571	0.5000	1.1053
6	4	0.0902	0.9473	0.4000	1.0381
7	5	0.0361	0.9834	0.3333	1.0122

Similarly, to demonstrate the log-concave properties of cold standby redundancy, the same 2-out-of- n configuration is considered. However, for the cold standby systems, the dormant failure rate is set to zero: $\lambda_d = 0$. The system reliability and its improvement factors are shown in Table 20.2. The results demonstrate that both $\frac{\delta_{i+1}}{\delta_i}$ and $\frac{R(n+1)}{R(n)}$ are decreasing functions even for the cold standby redundancy.

20.5.3 Optimal Allocation of Fixed Number of Total Spares

Consider a special case of the optimization problem where the total number of spares that are allocated to all subsystems is fixed. Further, consider that there is a fixed budget for procuring the spares and the cost (weight or volume) of each spare is the same. The objective is to find the optimal allocation of spares at each subsystem that maximizes the overall system reliability. From a mathematical point of view, it is a nonlinear integer programming problem. Prior to the publication of [23,33] there were no efficient solutions to find the global optimal allocation spares when subsystems were modeled using k -out-of- n cold/warm standby redundancy.

The system reliability can be obtained from (20.57). Further, the reliability of each subsystem can be expressed as a product of different factors. Specifically,

$$R_i(k_i, n_i) = R_i(k_i, k_i) \prod_{j=0}^{n_i-k_i} \phi_i(j)$$

$$\phi_i(j) = \frac{R_i(k_i, k_i + j)}{R_i(k_i, k_i + j - 1)} \quad (20.58)$$

The number of $\phi_i(j)$ terms in the product is equal to the number of spares in the subsystem. For example, consider a 2-out-of-4 warm standby configuration for a specific subsystem (e.g., subsystem 1). The subsystem reliability can be expressed as

$$R_1(2, 4) = R_1(2, 2) \cdot \frac{R_1(2, 3)}{R_1(2, 2)} \cdot \frac{R_1(2, 4)}{R_1(2, 3)} \quad (20.59)$$

From the log-concave properties of warm standby systems, $\phi_i(j)$ is known to be a decreasing function in j for each i (subsystem). When no spares are added to the subsystems, the system reliability is

$$R_{sys} = \prod_{i=1}^M R_i(k_i, k_i) \quad (20.60)$$

Now assume that s_i spares have already been added to subsystem- i and the total number of spares added to all subsystems is S . In other words, $S = \sum_{i=1}^M s_i$. Hence, the system reliability is

$$R_{sys} = \left[\prod_{i=0}^M R_i(k_i, k_i) \right] \cdot \left[\prod_{i=1}^M \prod_{j=1}^{s_i} \phi_i(j) \right] \quad (20.61)$$

The product term in the first set of square brackets is the same for any given set of spares allocation. Hence, it can be considered as constant. Therefore, system reliability can be maximized by maximizing the product term in the second set of square brackets. The number of terms in the second set of square brackets is equal to the total number of spares (S) added to the system. If this term includes only the highest possible $\phi_i(j)$ values, then the system reliability is maximized. Alternatively, if all possible values of $\phi_i(j)$ can be arranged in a descending order and we were able to select the first S values, the system reliability can be maximized. However, such a selection is valid only if, for any given subsystem- i , the factor corresponding to the j th spare can appear only if the factor corresponding to $(j - 1)$ th spare is already added. Mathematically, $\phi_i(j)$ can be included only if $\phi_i(j - 1)$ has been already included. However, such a selection is possible for the warm standby system due to its log-concave property. Therefore, it plays a key role in finding the solution to optimal allocation problem.

Note that $\phi_i(j - 1) > \phi_i(j)$ for warm standby system. Hence, $\phi_i(j - 1)$ will be included in the list before including $\phi_i(j)$. Further, due to the log-concave property, it is easy to identify the top $\phi_i(j)$ values without explicitly sorting them, because $\phi_i(j)$ is a decreasing function in j for each i . Hence, the highest value among all $\phi_i(j)$ is corresponding to the highest value among $\phi_i(1)$ for $i \in \{1, 2, \dots, M\}$. Let $\phi_i(1)$ be the current set of $\phi_i(j)$ values. It has M elements. Let the

corresponding subsystem that has the highest value of $\phi_i(1)$ be subsystem- m . To find the next highest $\phi_i(j)$ value, only the m^{th} element of the current set of $\phi_i(j)$ values needs to be updated with $\phi_m(2)$. This selection procedure can be continued until the maximum limit on S , i.e., (S_{\max}) , is reached.

Algorithm-C: Optimal Redundancy Allocation for Fixed Number of Spares

The detailed steps of the proposed algorithm are shown below:

- Step 1: **Set** $(s_1^*, \dots, s_M^*) = (0, \dots, 0)$ and $v = 0$.
- Step 2: **If** $(v < S_{\max})$ **then goto** step 3
 else take $s^* = (s_1^*, \dots, s_M^*)$ **as the optimal allocation;**
 Stop
 End If
- Step 3: **Find** m such that

$$\phi_m(s_i^* + 1) = \max_{1 \leq i \leq M} [\phi_i(s_i^* + 1)]$$
- Step 4: **Set** $s_m^* = s_m^* + 1$: increase the allocation to subsystem m by 1 and
 Set $v = v + 1$. **Go to** step 2

The above algorithm can be explained in simple terms. It adds a spare to the system one at a time until the allowed limit is reached. At each step, it allocates a spare to one of its subsystems where the percentage of improvement is the highest. This allocation algorithm is nothing but a simple greedy algorithm based on the steepest increment method. In other words, the allocation algorithm is not only simple but

also leads to global optimal solution in spite of its greedy nature. Note that the optimal allocation method is not the same as the method that improves the least reliable subsystem or allocates a spare to a subsystem where the improvement itself is the highest. The key factor is to allocate the spares to a subsystem where *reliability improvement percentage is the highest* rather than improving the least reliable subsystem or adding spares to a subsystem where the reliability improvement is the highest.

20.5.4 Optimal Allocation with Different Resource Consumption Rates

The allocation method described in Sect. 20.5.3 can also be extended to the case where different spares have different resource consumption rates (costs). In this chapter, we assume that the resource consumption rates are integers. When the consumption rates are real, in most cases, an appropriate multiplication factor can be used to convert them into integers. Further, the cost of spares can be specified in dollars or cents. However, to improve computation efficiency, it is recommended to represent them as the smallest possible integers. For example, if the costs of spares are in thousands of dollars, then the resource consumption rates can be represented in units of thousand dollars.

Algorithm-D: Optimal Redundancy Allocation for Spares with Different Resource Consumption

Let c_i be the cost of a spare (or resource consumption rate) used in subsystem- i . Let C be the total allowed budget for procuring the spares. Note that the proposed method computes the optimal allocation of spares for each integer value of budget B where $B \leq C$. The algorithm to find the optimal allocation of spares is

- Step 1: **Set** $s_1^*(0) = 0, \dots, s_M^*(0) = 0$ and $B = 0$.
- Step 2: **Compute** $R^*(0)$ using $s^*(0)$.
- Step 3: **If** $(B < C)$ **then go to** step 4
 else take $s^* = (s_1^*, \dots, s_M^*)$ **as the optimal allocation; Stop**
 EndIf
- Step 4: **Find** m such that

$$m = \operatorname{argmax}_{1 \leq i \leq M} [R^*(B - c_i) \cdot \phi_i(s_i^*(B - c_i) + 1)]$$
- Step 5: **Set** $R^*(B)$ as:

$$R^*(B) = R^*(B - c_m) \cdot \phi_m(s_m^*(B - c_m) + 1)$$
- Step 6: **Set** $s^*(B) = s^*(B - c_m)$.
 Then update $s^*(B) = s^*(B - c_m)$: increase the allocation to subsystem m by 1 and
 Set $B = B + 1$. **Go to** step 3

20.5.5 Numerical Example

Example 5 Consider a system consisting of five subsystems (I, II, III, IV, and V) in a series. Subsystem-I uses cold standby; subsystems II, III, and IV use warm standby redundancy; and subsystem-V uses active redundancy. Table 20.4 shows the operational and dormant (standby) failure rates of components used in the subsystems and minimum number of components needed for the successful operation of the subsystems.

The objective is to allocate a total of 20 spares among all subsystems to maximize the system reliability for mission time $t = 1$. To demonstrate the allocation method discussed in Sect. 20.5.3, Table 20.4 shows the intermediate results, i.e., $R_i(k_i, k_i)$ and $\phi_i(j)$ values. The table also includes the ranks of $\phi_i(j)$ values as their subscripts for the range of ranks from 1 to 20, i.e., from 1 to S_{\max} (Table 20.3).

At the every step where the first spare is added, the current $\phi_i(j)$ values are nothing but the $\phi_i(1)$ values: [2.60, 1.58, **2.88**, 1.73, 1.54]. The largest value corresponds to subsystem-III. So, a spare is added to subsystem-III and then the $\phi_i(j)$ values are updated to [2.60, 1.58, 1.65, 1.73, 1.54]. Now the largest value corresponds to subsystem-I. So, the next spare is allocated to subsystem-I and then the $\phi_i(j)$ values are updated to [1.49, 1.58, 1.65, **1.73**, 1.54]. Hence, the next spare is added to subsystem-IV. Continuing the same procedure, the final optimal allocation of spares for different subsystems are obtained. The spares are added in the following order: {III, I, IV, III, II, V, I, III, IV, I, V, III, II, I, IV, III, V, II, III, I}, and the final allocation of the spares at subsystems [I, II, III, IV, V] are [5, 3, 6, 3, 3], and the corresponding system reliability is 0.9436.

Further, consider the generic case where the cost of spares is different for different subsystems. Let $[c_1, c_2, c_3, c_4, c_5]$ be [5, 6, 10, 7, 4]. The budget for procuring all spares is 140. Using the Algorithm-D in Sect. 20.5.4, the optimal

spares allocation is [5, 3, 6, 3, 4]. The system reliability corresponding to the optimal allocation is 0.9521. Note that storing all possible values of $s_i(j)$ is not needed. At any given step of the optimization, the algorithm needs to keep track of at most $\max(c_i)$ entries of $s_i(j)$ values for each combination of i and j . Hence, the storage requirement is $M \cdot \max(c_i)$, which is much lower than $M \cdot C$.

20.6 Conclusions

This chapter presents reliability analysis and optimal redundancy allocation of k -out-of- n warm standby redundancy with identical components following exponential failure time distributions. Closed-form expressions for various reliability characteristics and measures are presented. Specifically, expressions for system reliability, state probabilities, failure (hazard) rate, pdf, and MRL & MPL functions are presented in the forms that are similar to the case of active redundancy. Subsequently, the system reliability is expressed in several forms to gain new insights into the system reliability characteristics. The closed-form expressions and computational algorithms presented in this chapter produce robust and numerically stable system reliability evaluation. The analysis accounts for the effects of switch failures on demand on system reliability [9]. Further, it is proved that the time-to-failure distribution of k -out-of- n warm standby system is beta-exponential. Using this property, closed-form expressions for statistical measures of system failure time distribution are presented. Specifically, closed-form expressions are provided for moment-generating function, characteristic function, variance, higher order moments, skewness, kurtosis, and Rényi and Shannon entropies of the random variable representing the system failure time.

The system reliability improvement with an additional redundant component follows a negative binomial (Pólya) distribution and is log-concave. Further, it has shown that the system reliability function is log-concave n . Hence, the percentage improvement in system reliability with an additional spare decreases with the number of spares already added to the system. Using these log-concave properties, efficient linear-time computational algorithms for finding optimal system designs are presented. Because active redundancy is a special case of the warm standby redundancy, indirectly this chapter also provides some new results for the active redundancy case as well.

Table 20.3 Optimization problem parameters of the subsystems

Subsystem index: i	Redundancy type	$\lambda_{o,i}$	$\lambda_{d,i}$	k_i
I	Cold standby	0.8	0	2
II	Warm standby	0.6	0.06	1
III	Warm standby	0.5	0.125	4
IV	Warm standby	0.4	0.2	2
V	Active	0.2	0.2	3

Table 20.4 Intermediate results: $R_i(k_i, k_i)$ and $\phi_i(j)$ Values

Subsystem index: i	$R_i(k_i, k_i)$	$\phi_i(1)$	$\phi_i(2)$	$\phi_i(3)$	$\phi_i(4)$	$\phi_i(5)$	$\phi_i(6)$
I	0.202	2.600 ₂	1.492 ₇	1.176 ₁₀	1.059 ₁₄	1.018 ₂₀	1.004
II	0.549	1.582 ₅	1.118 ₁₃	1.025 ₁₈	1.005	1.0001	1.0001
III	0.135	2.880 ₁	1.652 ₄	1.278 ₈	1.122 ₁₂	1.019 ₁₆	1.019 ₁₉
IV	0.449	1.725 ₃	1.191 ₉	1.058 ₁₅	1.017	1.001	1.0013
V	0.549	1.544 ₆	1.127 ₁₁	1.034 ₁₇	1.009	1.001	1.0005

The exponential component lifetimes assumption made in this paper can be removed by extending the result to proportional hazards models subjected to arbitrary general baseline failure time distributions [50]. Similarly, in addition to switch failures on demand, the results can be extended to account for the failure effects of continuously monitoring switches on the system reliability [7, 9]. The same concepts used in this chapter can also be extended to analyze a k -out-of- n warm standby system with nonidentical components by considering different paths that lead the system failure. Similar to nonrepairable case, the simple formulas presented in this chapter can also be extend to k -out-of- n warm standby system with identical but repairable components. The results presented in this chapter can easily be extended to find the optimal cost-effective designs that consider the cost of spares as well as cost of system failure [41]. We consider these extensions as a future research work.

Acknowledgments The first author of this book chapter, Suprasad Amari, would like to acknowledge that the joint research work presented in this chapter was conducted prior to his joining at the BAE Systems and the opinions expressed in this article are the author's own and do not reflect the views of the BAE Systems. The first author also would like to remember and acknowledge his doctoral research advisor, Professor Ravindra B. Misra, for introducing him to the reliability model discussed in this chapter.

References

- ANSI/IEEE 446-1995: IEEE Recommended Practice for Emergency and Standby Power Systems for Industrial and Commercial Applications (IEEE Orange Book). IEEE, Piscataway (1995)
- Sklaroff, J.R.: Redundancy management technique for space shuttle computers. IBM J. Res. Dev. **20**(1), 20–28 (1976)
- Pham, H., Phan, H.K., Amari (Suprasad), S.V.: A general model for evaluating the reliability of telecommunications systems. Commun. Reliab. Maintainab. Supportab. Int. J. **2**, 4–13 (1995)
- She, J., Pecht, M.G.: Reliability of a k -out-of- n warm-standby system. IEEE Trans. Reliab. **41**(1), 72–75 (1992)
- Kuo, W., Prasad, V.R., Tillman, F.A., Hwang, C.L.: Optimal Reliability Design: Fundamentals and Applications. Cambridge University Press, Cambridge (2001)
- Amari, S.V., Zuo, M.J., Dill, G.: $O(kn)$ algorithms for analyzing repairable and non-repairable k -out-of- n : G systems. In: Misra, K.B. (ed.) Handbook of Performability Engineering, Chap. 21, pp. 309–320. Springer, Berlin (2008)
- Kuo, W., Zuo, M.J.: Optimal Reliability Modeling: Principles and Applications. John Wiley & Sons, Hoboken (2003)
- Asadi, M., Goliforushani, S.: On the mean residual life function of coherent systems. IEEE Trans. Reliab. **57**(4), 574–580 (2008)
- Amari, S.V., Dill, G.: A new method for reliability analysis of standby systems. In: Proceedings of the Annual Reliability and Maintainability Symposium 13D–1 (2009)
- Yearout, R.D., Reddy, P., Grosh, D.L.: Standby redundancy in reliability – a review. IEEE Trans. Reliab. **35**(3), 285–292 (1986)
- Gnedenko, B.V., Belayaev, Y.K., Solovyeb, A.D.: Mathematical Methods of Reliability Theory. Academic Press, Cambridge (1969)
- Li, X., Yan, R., Zuo, M.J.: Evaluating a warm standby system with components having proportional hazard rates. Oper. Res. Lett. **37**, 56–60 (2009)
- Morrison, D.F., David, H.A.: The life distribution and reliability of a system with spare components. Ann. Math. Stat. **31**(4) (1960)
- Barlow, R.E., Proschan, F.: Mathematical Theory of Reliability. SIAM, Philadelphia (1996)
- Amari, S.V.: Reliability analysis of k -out-of- n cold standby systems with Erlang distributions. Int. J. Performab. Eng. **8**(4), 417–425 (2012)
- Amari, S.V.: Reliability of k -out-of- n standby systems with gamma distributions. Proc. Ann. Reliab. Maintainab. Symp., 1–6 (2012)
- Mohammad, R., Amari, S.V., Kalam, A.: Reliability of k -out-of- n cold standby systems with Rayleigh distributions. In: Proceedings of the 18th ISSAT International Conference on Reliability and Quality in Design, pp. 112–119 (2012)
- Eryilmaz, S.: Reliability of a k -out-of- n system equipped with a single warm standby component. IEEE Trans. Reliab. **62**(2), 499–503 (2013)
- Amari, S.V., Dill, G.: Redundancy optimization problem with warm-standby redundancy. In: Proceedings of the Annual Reliability and Maintainability Symposium, pp. 1–6 (2010)
- Levitin, G., Xing, L., Dai, Y.: Cold standby systems with imperfect backup. IEEE Trans. Reliab. **65**(4), 1798–1809 (2015)
- Wang, W., Xiong, J., Xie, M.: Cold-standby redundancy allocation problem with degrading components. Int. J. Gen. Syst. **44**(7–8), 876–888 (2015)
- Tannous, O., Xing, L., Peng, R., Xie, M.: Reliability of warm-standby systems subject to imperfect fault coverage. Proc. Inst. Mech. Eng. O J. Risk Reliab. **228**(6), 606–620 (2014)
- Amari, S.V., Pham, H., Misra, R.B.: Reliability characteristics of k -out-of- n warm standby systems. IEEE Trans. Reliab. **61**(4), 1007–1018 (2012)
- Amari, S.V., Wang, C., Xing, L., Mohammad, R.: An efficient phased-mission reliability model considering dynamic k -out-of- n subsystem redundancy. IISE Trans. **50**(10), 868–877 (2018)
- Levitin, G., Xing, L., Ben-Haim, H., Huang, H.-Z.: Dynamic demand satisfaction probability of consecutive sliding window systems with warm standby components. Reliab. Eng. Syst. Saf. **189**, 397–405 (2019)
- Levitin, G., Xing, L., Dai, Y.: Optimal sequencing of warm standby elements. Comput. Ind. Eng. **65**(4), 570–576 (2013)
- Boddu, P., Xing, L., Levitin, G.: Sequencing optimization for k -out-of- n : G cold-standby systems considering reliability and energy consumption. In: Ram, M., Dohi, T. (eds.) Systems Engineering: Reliability Analysis Using k -out-of- n Structures, Chap. 5, pp. 85–104. CRC Press: Taylor & Francis Group Company, Boca Raton (2019)
- Levitin, G., Xing, L., Haim, H.B., Dai, Y.: Optimal structure of series system with 1-out-of- n warm standby subsystems performing operation and rescue functions. Reliab. Eng. Syst. Saf. **188**, 523–531 (2019)
- Levitin, G., Xing, L., Luo, L.: Influence of failure propagation on mission abort policy in heterogeneous warm standby systems. Reliab. Eng. Syst. Saf. **183**, 29–38 (2019)
- Levitin, G., Xing, L., Dai, Y., Vokkarane, V.M.: Dynamic check-pointing policy in heterogeneous real-time standby systems. IEEE Trans. Comput. **66**(8), 1449–1456 (2017)
- Jia, H., Levitin, G., Ding, Y., Song, Y.: Reliability analysis of standby systems with multi-state elements subject to constant transition rates. Qual. Reliab. Eng. Int. **35**(1), 318–328 (2019)
- Amari, S.V., Pham, H.: A new insight into k -out-of- n warm standby model. Int. J. Performab. Eng. **6**(6), 615–617 (2010)
- Amari, S.V.: Optimal redundancy allocation using log-concave properties. In: Proceedings of the Annual Reliability and Maintainability Symposium, 01C (2014)

34. Nadarajah, S., Kotz, S.: The beta exponential distribution. *Reliab. Eng. Syst. Saf.* **91**, 689–697 (2006)
35. Amari, S.V., Misra, R.B.: Closed-form expressions for distribution of sum of exponential random variables. *IEEE Trans. Reliab.* **46**(4), 519–522 (1997)
36. DeGroot, M.H.: *Optimal Statistical Decisions*, WCL edn. Wiley-Interscience, Hoboken (2004)
37. Abramowitz, A., Stegun, I.A.: *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, 10th Printing edn. National Bureau of Standards, Gaithersburg (1972)
38. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: *Numerical Recipes: The Art of Scientific Computing*, 3rd edn. Cambridge University Press, Cambridge (2007)
39. Amari, S.V., Dugan, J.B., Misra, R.B.: A separable method for incorporating imperfect fault-coverage into combinatorial models. *IEEE Trans. Reliab.* **48**(3), 267–274 (1999)
40. Amari, S.V., Pham, H., Dill, G.: Optimal design of k -out-of- n : G subsystems subjected to imperfect fault-coverage. *IEEE Trans. Reliab.* **53**(4), 567–575 (2004)
41. Pham, H.: On the optimal design of k -out-of- n : G subsystems. *IEEE Trans. Reliab.* **41**(4), 572–574 (1992)
42. Trivedi, K.: *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*. John Wiley & Sons, Hoboken (2001)
43. Nanda, A.K., Sengupta, D.: Discrete life distributions with decreasing reversed hazard. *Sankhyā Indian J. Stat.* **67**, 106–125 (2005)
44. Prasad, V.R., Kuo, W., Oh Kim, K.M.: Optimal allocation of s -identical, multi-functional spares in a series system. *IEEE Trans. Reliab.* **48**(2), 118–126 (1999)
45. Amari, S.V.: Addendum to: generic rules to evaluate system-failure frequency. *IEEE Trans. Reliab.* **51**(3), 378–379 (2002)
46. Bagnoli, M., Bergstrom, T.: Log-concave probability and its applications. *Eco. Theory* **26**(2), 445–469 (2005)
47. Reijns, G., van Gemund, A.: Reliability analysis of hierarchical systems using statistical moments. *IEEE Trans. Reliab.* **56**(3), 525–533 (2007)
48. Tavangar, M., Asadi, M.: A study on the mean past lifetime of the components of $(n - k + 1)$ -out-of- n system at the system level. *Metrika* **72**(1), 59–73 (2009)
49. Ortega, E.-M.: A note on some functional relationships involving the mean inactivity time order. *IEEE Trans. Reliab.* **58**(1), 172–178 (2009)
50. Amari, S.V., Misra, K.B., Pham, H.: Tampered failure rate load-sharing systems: status and perspectives. In: Misra, K.B. (ed.) *Handbook of Performability Engineering*, Chap. 20, pp. 291–308. Springer, Berlin (2008)



Suprasad V. Amari received his PhD from the Indian Institute of Technology, Kharagpur, India, in 1998. He worked at the Tata Consultancy Services, in India, and at the Relex Software Corporation, at the Parametric Technology Corporation, and at the BAE Systems in US. He works now on system engineering methodologies, focused on reliability modeling aspects of defense systems.



Dr. Hoang Pham is a Distinguished Professor and former Chairman (2007-2013) of the Department of Industrial and Systems Engineering at Rutgers University, New Jersey. He is the author or coauthor of 6 books and has published over 180 journal articles and edited 15 books. He is a Fellow of the IEEE and IIE.



Importance and Sensitivity Analysis on Complex Repairable Systems and Imprecise System Reliability

21

Geng Feng

Contents

21.1	Introduction	403
21.2	Relationship Between Structure Function and Survival Signature	404
21.3	Importance and Sensitivity Analysis on Repairable System	405
21.3.1	Reliability Analysis on Repairable System.....	405
21.3.2	Importance Measure of a Specific Component.....	406
21.3.3	Importance Measure of a Set of Components.....	406
21.3.4	Quantify Importance Degree.....	408
21.4	Imprecision Within Survival Signature	408
21.5	Numerical Examples	408
21.5.1	Importance and Sensitivity Analysis.....	408
21.5.2	Imprecision Within the System.....	411
21.6	Conclusions	412
	References	412

In order to address the above questions, the efficient simulation approaches based on structure function and survival signature have been proposed respectively to analyze the complex repairable systems. Based on this, component importance index has been introduced to perform sensitivity analysis on a specific component or a set of components within the repairable system. In addition, the proposed simulation method can be used to deal with imprecision within the survival signature. Numerical examples are presented to show the applicability of the above approaches.

Keywords

Importance measures · Sensitivity analysis · Complex systems · Survival signature · Reliability · Repairable system

Abstract

The identification of the components which are responsible for the performance of system is a key task to improve the safety and reliability. However, such analysis is difficult and challenging when large and complex systems are analyzed. The survival signature which is presented recently cannot only hold the merits of the former system signature but is efficient to deal with complex system with multiply component types. In real engineering applications, components can be repaired after failure. Hence, it is essential to identify which component or component set is most critical the complex repairable system. What is more, due to lack of data or the confidential information, it is difficult to know the full configuration of the system, which leads to an imprecise survival signature.

21.1 Introduction

System signature has been introduced recently by Samaniego [25]; it can separate the system structure from the components failure time distributions, which makes it efficient to quantify the reliability of the system. However, system signature has a very important assumption that all components in the coherent system should be in a single type, which makes it intractable to analyze complex system with two or more component types.

In order to overcome the limitations of the system signature, [7] proposed the survival signature, which not only reserves the merits of the system signature but can be applied to analyze system with multiple component types. In essence, it does not have the assumption that components of different types are exchangeable, which overcomes the long-standing limitation of the system signature. This is useful when a system may have components with failure times that follow different probability distributions [8]. Therefore, survival

G. Feng (✉)
School of Engineering, University of Central Lancashire, Preston, UK
e-mail: gfeng@uclan.ac.uk

signature is a promising method for application to complex systems and networks.

Recently, Aslett developed an R package to calculate the survival signature [1], and an efficient algorithm for the exact computation of the survival signature of large systems was put forward by Reed [24]. A nonparametric predictive inference for system reliability using the survival signature was proposed by Coolen et al. [11]. Based on the above concepts, Feng et al. developed an analytical and numerical method to perform reliability analysis on systems with uncertain components parameters [14]. These survival signature-based methods are efficient for non-repairable system reliability analysis; however, they do not take the complex repairable system into consideration. Simulation methods has attractive features for analyzing complex and large systems [22], which makes it possible to deal with the repairable complex systems reliability. Survival signature methodology is extended to analyze systems with common cause failures [13, 18].

The risks are unavoidable, and as such the key challenge in engineering risk analysis is to identify the components of the system that contribute most to risk [20]. Importance and sensitivity analysis is an essential way to find out the most critical component, which helps engineers to allocate the limited resources. Therefore, importance and sensitivity analysis is a key point in tracing bottlenecks in systems and in identifying the most important component [19]. The definition of component importance measure is first introduced by Birnbaum [3], which is obtained by partial differentiation of the system reliability with respect to the given component reliability. An improvement or decline in reliability of the component with the highest importance will cause the greatest increase or decrease in system reliability. Based on this achievement, many other component importance measures have been introduced, e.g., Fussell-Vesely importance measure [17, 27], failure criticality index [28], structure importance measure [4], risk reduction worth, and risk achievement worth [5]. The aforementioned importance measures mainly focus on an individual component of a non-repairable system; however, it is important to calculate the importance degree of component within a repairable system. What is more, it is of interest to evaluate the importance of a set of components or a subsystem with a few components in many practical situations.

Probabilistic uncertainty and imprecision in structural parameters and in environmental conditions and loads are challenging phenomena in engineering analysis [2]. There is a wide variety of reasons why imprecise probability might be of particular relevance in the area of system reliability [6, 23]. For example, due to limited data or confidentiality for the system, it sometimes cannot get the precise survival signature. In order to deal with imprecision, Ferson et al. introduced the probability box, which is also called p-box, to define the set of the imprecise values [15, 16]. Coolen

and Coolen-Maturi [9] linked the (imprecise) probabilistic structure function to the survival signature. The interval of the survival signature will lead to imprecise survival function of the system.

The structure function-based method and the survival signature-based method are both based on the Monte Carlo simulation, which is general and useful for many problems. By generating the state evolution of each component, the structure function is computed to determine the state of the system, while survival signature is a summary of the structure signatures, which is efficient to deal with the complex configuration systems.

A survival signature-based method is proposed in this paper, which is efficient to analysis repairable system reliability. This is essential when dealing with complex repairable systems since they can only be analyzed through simulation method. In order to find out the most “critical” component in the system, new component importance measures which based on survival signature are introduced to analysis individual component and component sets respectively. What is more, a new relative criticality index is used to quantify the importance degree of the component. There exists imprecise survival signature if the full configuration of the system is unknown or only uncertain survival signature is known due to confidential contract, it is essential to deal with the imprecision by simulation methods. Therefore, the intervals of the survival function of the complex system can be gotten. The applicability of the proposed approach is demonstrated by solving the numerical examples.

This paper is organized as follows. Section 21.2 gives a conception and relationship between the structure function and survival signature. The relative importance index and quantitative importance index to perform importance and sensitivity analysis on complex repairable systems are proposed in Sect. 21.3. Section 21.4 shows how to deal with the imprecise survival signature. The applicability and performance of the proposed approaches are presented in Sect. 21.5. Finally Sect. 21.6 closes the paper with conclusions.

21.2 Relationship Between Structure Function and Survival Signature

For a system with m components, let state vector $\underline{x} = (x_1, x_2, \dots, x_m) \in \{0, 1\}^m$, with $x_i = 1$ if the i th component works and $x_i = 0$ if not. The structure function $\phi(\underline{x}) : \{0, 1\}^m \rightarrow \{0, 1\}$, defined for all possible \underline{x} , takes the value 1 if the system functions and 0 if not for state vector \underline{x} . In this paper, attention is restricted to coherent systems, for which $\phi(\underline{x})$ is not decreasing in any of the components of \underline{x} , so system functioning cannot be improved by worse performance of one or more of its components. It is further

assumed that $\phi(0) = 0$ and $\phi(1) = 1$, so the system fails if all its components fail and functions if all its components function.

Now consider a system with $K \geq 2$ types of components, with m_k components of type $k \in \{1, 2, \dots, K\}$ and $\sum_{k=1}^K m_k = m$. Assume that the random failure times of components of the system type are exchangeable, while full independence is assumed for components belong to different types. The components of the same type can be grouped together due to the arbitrary ordering of the components in the state vector, which leads to a state vector can be presented as $\underline{x} = (\underline{x}^1, \underline{x}^2, \dots, \underline{x}^K)$, with $\underline{x}^k = (x_1^k, x_2^k, \dots, x_{m_k}^k)$ illustrating the states of the components of type k . Coolen [7] introduced the survival signature for such a system, denoted by $\Phi(l_1, l_2, \dots, l_K)$, with $l_k = 0, 1, \dots, m_k$ for $k = 1, 2, \dots, K$, which is defined to be the probability that the system functions given that l_k of its m_k components of type k work, for each $k \in \{1, 2, \dots, K\}$.

There are $\binom{m_k}{l_k}$ state vectors \underline{x}^k with $\sum_{i=1}^{m_k} x_i^k = l_k$ ($k = 1, 2, \dots, K$), and let S_{l_1, l_2, \dots, l_K} denote the set of all state vectors for the whole system. Due to independent and identical distributed (*iid*) assumption, all the state vectors $\underline{x}^k \in S_{l_k}^k$ are equally likely to occur, so the survival signature can be written as:

$$\Phi(l_1, \dots, l_K) = \left[\prod_{k=1}^K \binom{m_k}{l_k}^{-1} \right] \times \sum_{\underline{x} \in S_{l_1, \dots, l_K}} \phi(\underline{x}) \quad (21.1)$$

It can be seen from Eq. 21.1 that survival signature is a summary of structure functions. Let $C_k(t) \in \{0, 1, \dots, m_k\}$ denote the number of k components working at time t . Assume that the components of type k have a known cumulative distribution function (*CDF*) $F_k(t)$ and the component failure times of different types are independent; then:

$$\begin{aligned} P\left(\bigcap_{k=1}^K \{C_k(t) = l_k\}\right) &= \prod_{k=1}^K P(C_k(t) = l_k) \\ &= \prod_{k=1}^K \binom{m_k}{l_k} [F_k(t)]^{m_k - l_k} [1 - F_k(t)]^{l_k} \end{aligned} \quad (21.2)$$

Hence, the survival function of the system with K types of components becomes:

$$P(T_s > t) = \sum_{l_1=0}^{m_1} \dots \sum_{l_K=0}^{m_K} \Phi(l_1, \dots, l_K) P\left(\bigcap_{k=1}^K \{C_k(t) = l_k\}\right) \quad (21.3)$$

Equation 21.3 shows that the survival signature can separate the structure of the system from the failure time distribution of its components, which is the main advantage of the system signature. What is more, the survival signature only

needs to be calculated once when perform reliability analysis on the same system. The survival signature is closely related with system signature. For a special case of a system with only one type ($K = 1$) of components, the survival signature and the Samaniego's system signature are directly linked to each other through a simple equation [7]; however, the latter cannot be easily generalized for systems with multiple types ($K \geq 2$) of components. As a result, the survival signature cannot only hold the merits of system signature but is efficient for complex system with multiple components types.

21.3 Importance and Sensitivity Analysis on Repairable System

Component importance measures and sensitivity analysis is evaluable in real engineering world. The existing importance measures are mostly calculated through analytical approaches, and application of these measures to complex repairable systems may be intractable. In order to overcome intractability, two important measures work by figuring out how much each component or a component set contribute to system unavailability. What is more, another index is used to quantify the importance degree of the specific component and component set. It always takes the system with m components which belong to K component types, for example, in this section.

21.3.1 Reliability Analysis on Repairable System

If a system with m components is repairable [12], the state transition diagram of a component is presented in Fig. 21.1. The schematic diagram of the m components status and the corresponding system performance can be seen in Fig. 21.2.

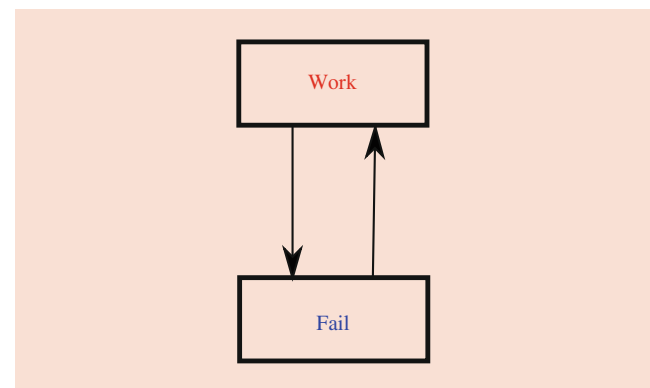


Fig. 21.1 State transition diagram of a component

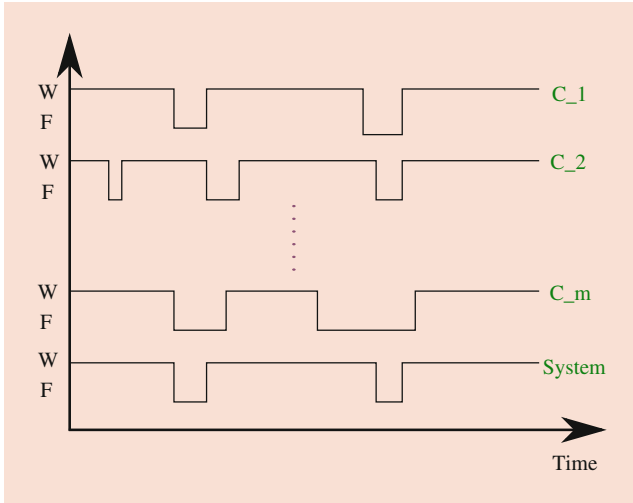


Fig. 21.2 Schematic diagram of the m components status and the corresponding system performance

The reliability analysis on repairable systems can turn to structure function-based method and survival signature-based method, respectively.

For the structure function-based method, it is essential to estimate the system status at each critical time points, which means the beginning time for each component fails and the its finish time to repair.

While for the second repairable system reliability analysis method, it is based on the survival signature, and it uses the system “production level” idea that is proposed in [29].

The flowchart of the repairable system survival function is evaluated by structure function, and survival signature can be seen in Fig. 21.3. These algorithms can be obtained from the approach proposed by one of the authors in [29].

21.3.2 Importance Measure of a Specific Component

The relative importance index $RI_i(t)$ of the i th component at time t that first used in [14] can be extended to analyze the importance degree of components in the repairable system. To be specific, it is the repairable system survival function probability differences if the i th component works or not. The mathematical expression formula of the relative importance index which is based on survival signature can be expressed as:

$$RI_i^{SS}(t) = P(T_S > t | T_i > t) - P(T_S > t | T_i \leq t) \quad (21.4)$$

where $P(T_S > t | T_i > t)$ represents the probability that the repairable system works knowing that the i th component functions; $P(T_S > t | T_i \leq t)$ denotes the repairable system survival probability if the same component fails.

However, since the relative importance index calculated by survival signature needs to know the survival signature of the new system. If the system is complex, it is a time-consuming work. Therefore, the structure function-based relative importance index has been introduced to identify the specific component’s importance degree; its equation is:

$$RI_i^{SF}(t) = P(T_S > t | x_i \text{ repairable}) - P(T_S > t | x_i \text{ non-repairable}) \quad (21.5)$$

where $P(T_S > t | x_i \text{ repairable})$ means the survival function of the repairable system if the i th component can be repaired normally, while $P(T_S > t | x_i \text{ non-repairable})$ indicates the probability that the system functions knowing that the same component cannot be repaired after failure.

21.3.3 Importance Measure of a Set of Components

It is sometimes important to evaluate the importance of a set of components instead of a specific one in the real engineering world. Therefore, the relative importance index for a specific component can be extended to component set of k , which can be denoted by $RI_k(t)$. The set of components can be either in one single type or different types.

For the first situation, $RI_k(t)$ is convenient to combine with the survival signature, and it is the probability difference values of the repairable system works if the components of type k are repairable or they cannot be repaired. The expression can be written as follows:

$$RI_k^{SS}(t) = P(T_S > t | l_k \text{ repairable}) - P(T_S > t | l_k \text{ non-repairable}) \quad (21.6)$$

where $P(T_S > t | l_k \text{ repairable})$ represents the probability that the repairable system works if components of type k are repairable; $P(T_S > t | l_k \text{ non-repairable})$ denotes the probability that the repairable system functions knowing that the components of type k cannot be repaired.

While for the second condition, it is more efficient to analyze the importance degree of a set of components that belong to different types. The mathematical equation of the structure function-based method is:

$$RI_k^{SF}(t) = P(T_S > t | \underline{x}_i^k \text{ repairable}) - P(T_S > t | \underline{x}_i^k \text{ non-repairable}) \quad (21.7)$$

where $P(T_S > t | \underline{x}_i^k \text{ repairable})$ means the survival probability of the repairable system if the set of the i th component of type k is repairable; here $i \in (1, 2, \dots, m_k)$ and $k \in$

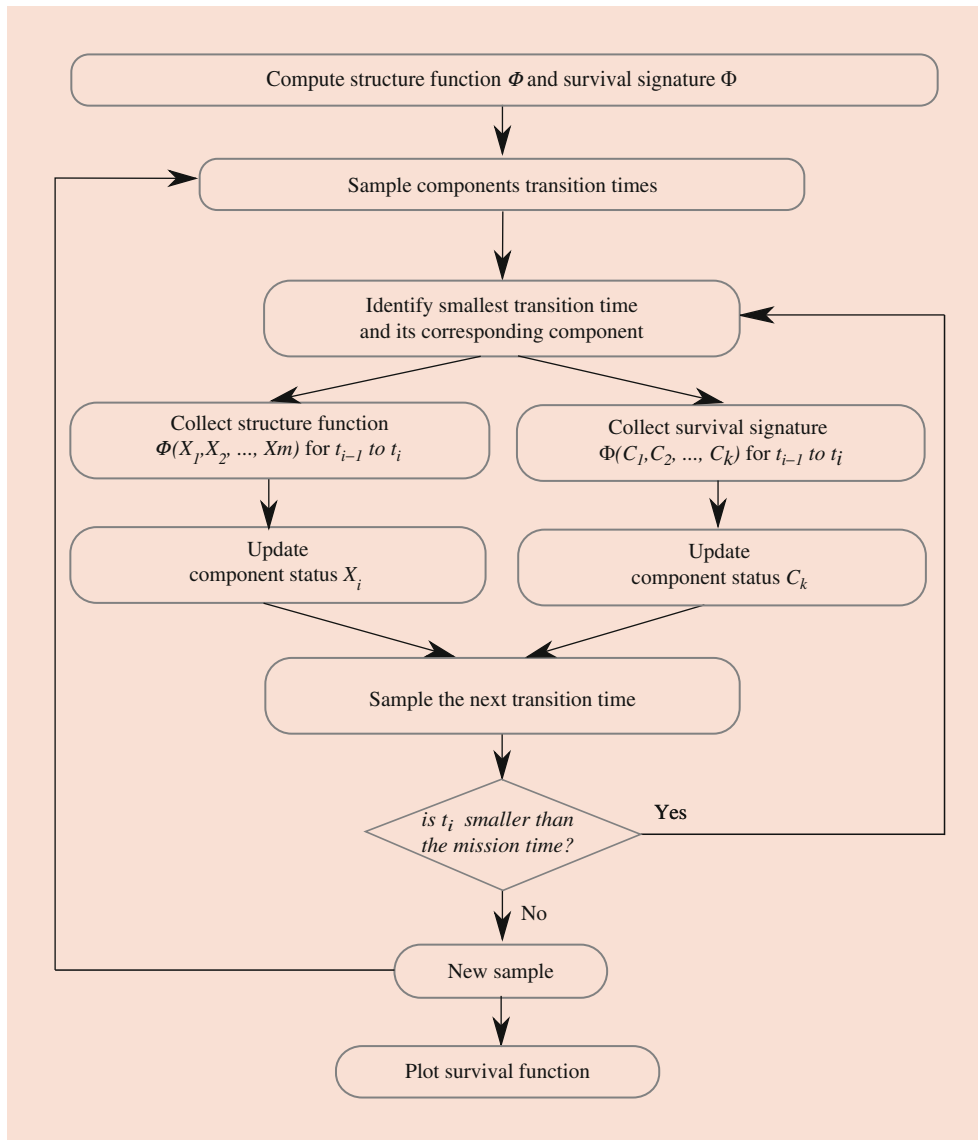


Fig. 21.3 Flowchart of structure function and survival signature based survival function evaluation method for repairable system

$(1, 2, \dots, K)$. $P(T_S > t \mid \underline{x}_i^k \text{ non-repairable})$ represents the survival function of the system given that the i th component of type k set is non-repairable.

It can be seen that both $RI_i(t)$ and $RI_k(t)$ are time-dependent and both of them can be calculated by survival signature-based method and structure function-based method, respectively. What is more, they reveal the trend of the importance degree of a specific component or a set of components within the repairable system during the survival time. The bigger the value of $RI_i(t)$ or $RI_k(t)$ is, the more “critical” is the i th component or the set of components on the repairable system reliability at a specific time t , and vice versa. This helps to allocate resources, which might

include resources for reliability improvement, surveillance and maintenance, design modification, security, operating procedure, training, quality control requirements, and a wide variety of other resource expenditures. By using the importance of a specific component or a set of components, resources expenditure can be properly optimized to reduce the total life cycle resource expenditures while keeping the risk as low as possible. In other words, for a given resource expenditure such as for maintenance, the importance measure of a specific component or set of components can be used to allocate resources to minimize the total system risk. This approach allows the risk manager to offer the “biggest bang for the buck” [21].

21.3.4 Quantify Importance Degree

In order to quantify importance degree of the specific component or a set of components, the quantitative importance index (*QI*) is introduced in this paper. The numerically obtained index for a repairable system is through Monte Carlo simulation method which is based on survival signature and structure function. The failure times of the system can be gotten through each trial; after having simulated many histories of the system, estimates are made of the desired relative criticality index statistically. For a system with *m* components belonging to *K* types, the quantitative importance index of the specific component *i* is expressed as:

$$QI_i = \frac{N_i^f}{\max\{N_1^f, \dots, N_i^f, \dots, N_m^f\}} \tag{21.8}$$

where N_i^f represents average number of system failures if the *i*th component cannot repair while the other components can be repaired normally; and $\max\{N_1^f, \dots, N_i^f, \dots, N_m^f\}$ denotes the biggest value of all N_i^f with $i = 1, 2, \dots, m$.

Similarly, for component set *k*, the *QI* can be written as:

$$QI_k = \frac{N_k^f}{\max\{N_1^f, \dots, N_k^f, \dots, N_K^f\}} \tag{21.9}$$

where N_i^f denotes the average failure number of the system if the component set *k* are non-repairable but the other component sets are repairable, while $\max\{N_1^f, \dots, N_k^f, \dots, N_K^f\}$ denotes the maximum number of all N_k^f with $k = 1, 2, \dots, K$.

These two indexes can quantify the importance degree of a specific component and a component set respectively, and the quantitative importance index values of all the components are compared with the biggest *QI* value. Therefore, the bigger the value is, the bigger influence of the *i*th component or the component set *k* on the repairable system.

21.4 Imprecision Within Survival Signature

Imprecise probability [10] generalizes classical probability in the sense that uncertainty about events are quantified via intervals, instead of single numbers. For example, the interval of an uncertain event *A* is $[\underline{P}(A), \overline{P}(A)]$ with $0 \leq \underline{P}(A) \leq \overline{P}(A) \leq 1$, where $\underline{P}(A)$ is called the lower probability for event *A*, $\overline{P}(A)$ is called the upper probability for event *A*, and $\Delta(A) = \overline{P}(A) - \underline{P}(A)$ is called the imprecision for event *A* [6].

As requirements for system quality have increased, the need for high system reliability is also increasing [26]. In the real industrial world, it is sometimes difficult to know the full configuration of the system if there has confidential contract of the company or only know the bounds of survival signature

due to lack of data. The interval of the survival signature can be presented by $[\underline{\Phi}(l_1, \dots, l_K), \overline{\Phi}(l_1, \dots, l_K)]$; therefore, the lower bound of the survival function of the system is:

$$\underline{P}(T_s > t) = \sum_{l_1=0}^{m_1} \dots \sum_{l_K=0}^{m_K} \underline{\Phi}(l_1, \dots, l_K) P\left(\bigcap_{k=1}^K \{C_k(t) = l_k\}\right) \tag{21.10}$$

and the corresponding upper bound of the survival function is:

$$\overline{P}(T_s > t) = \sum_{l_1=0}^{m_1} \dots \sum_{l_K=0}^{m_K} \overline{\Phi}(l_1, \dots, l_K) P\left(\bigcap_{k=1}^K \{C_k(t) = l_k\}\right) \tag{21.11}$$

It can be seen that the limited information about the survival signature can be used to derive bounds of the lower and upper survival function of the system.

21.5 Numerical Examples

Figure 21.4 shows the block diagram of a simplified auxiliary feed water system, which consists of fourteen components that belong to six component types, while Table 21.1 indicates the distribution types and parameters for failure process ($1 \rightarrow 2$) and repair process ($2 \rightarrow 1$) of all the components.

21.5.1 Importance and Sensitivity Analysis

Let's first perform the importance measure of a specific component which is based on structure function. The results can be seen in Fig. 21.5.

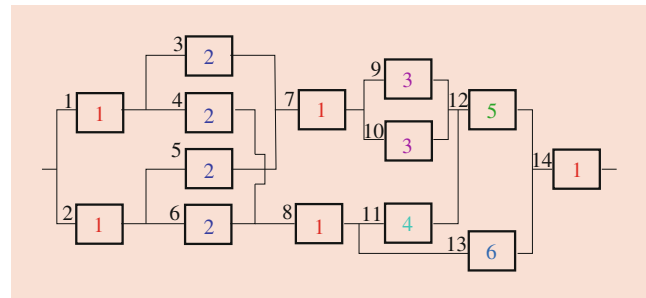


Fig. 21.4 The complex repairable system with six types of components. The numbers inside the component boxes indicate the component type. The numbers above the component boxes indicate the component indices

It is clear that component 14 always has higher relative importance index than the other 13 components, which means it is the most “critical” component in the repairable system. Then it comes to component 8. Component 13 has litter relative importance index values at the first time; however, its relative importance index values become bigger and bigger as time goes on, which just follows components 14 and 8. Component 1 and component 2 have similar relative importance values, which are sometimes crossover. The same circumstance occurs on components 4 and 6. The relative

importance of the five components (3, 5, 7, 9, 10) is always within 0.1, which means they have less importance influence degree than that of other components on the repairable system.

In the real application world, sometimes people want to know the importance degree of a set of components, i.e., the relative importance index of components of set 1 to set 6 in this repairable system. Figure 21.6 shows the results of them.

It can be seen that the relative importance index values of component sets 1 and 2 are bigger than that of other component sets; therefore, components of type 1 and 2 are more important than components of other types in this repairable system. On the contrary, component set 4 is the least important within the system because it has the smallest values of relative importance index. The values of component set 1 are higher than 2 at the beginning time; however, their values are the same as the survival time goes on. Component set 5 has lower relative importance values than component sets 3, but the values go up and rank the third within the six component sets in the last. Component sets 3 and 6 have the same relative importance values trend, although the value of set 5 is bigger than that of set 6 at the beginning time.

When it comes to analyzing the importance degree of a set of components which belong to different types, the efficient structure function method can be used. Suppose it is a necessary to perform sensitivity analysis on six component

Table 21.1 Distribution for failure and repair process of components

Component type	Process	Distribution	Parameters
1	1 → 2	Exponential	2.3
1	2 → 1	Uniform	(0.4,0.6)
2	1 → 2	Exponential	1.2
2	2 → 1	Uniform	(0.9,1.1)
3	1 → 2	Weibull	(1.7,3.6)
3	2 → 1	Uniform	(0.6,0.8)
4	1 → 2	Lognormal	(1.5,2.6)
4	2 → 1	Uniform	(1.0,1.2)
5	1 → 2	Weibull	(3.2,2.5)
5	2 → 1	Uniform	(1.2,1.4)
6	1 → 2	Gamma	(3.1,1.5)
6	2 → 1	Uniform	(1.1,1.3)

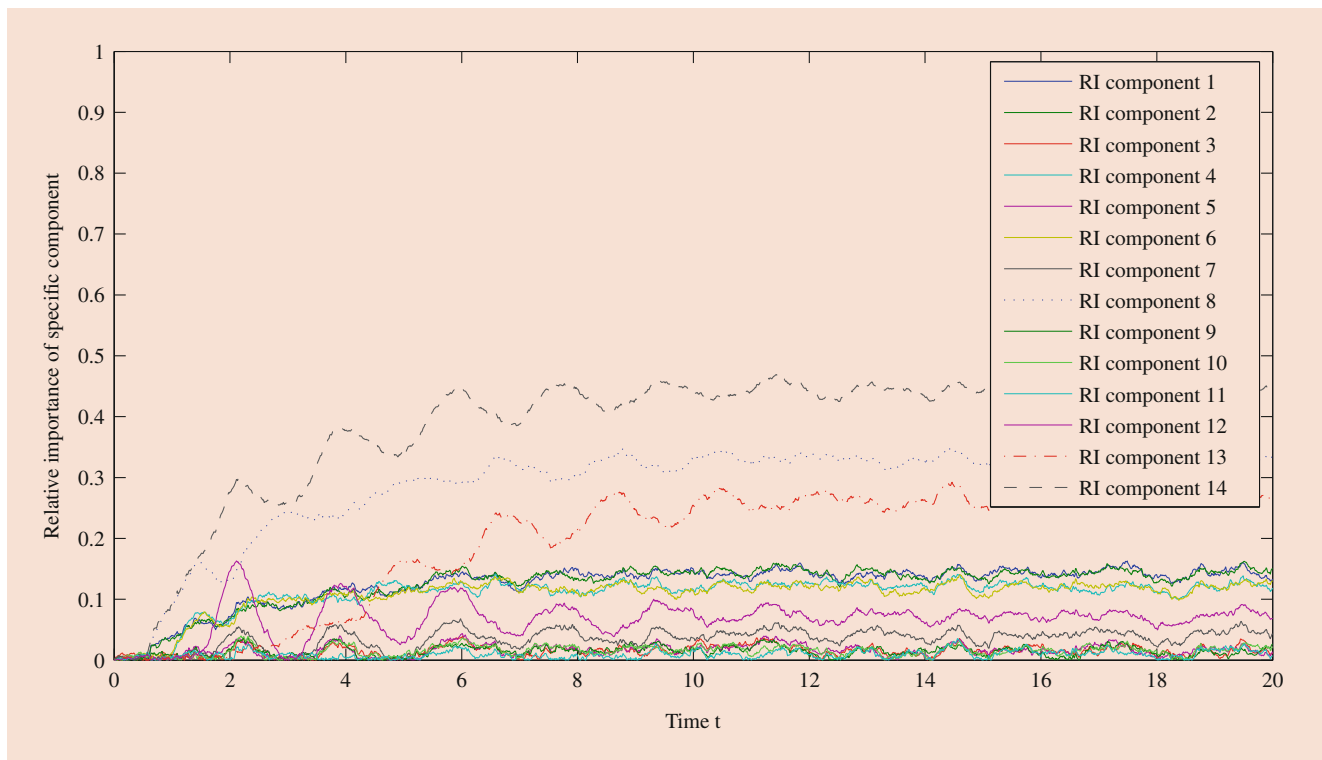


Fig. 21.5 Relative importance index of the specific component in the system

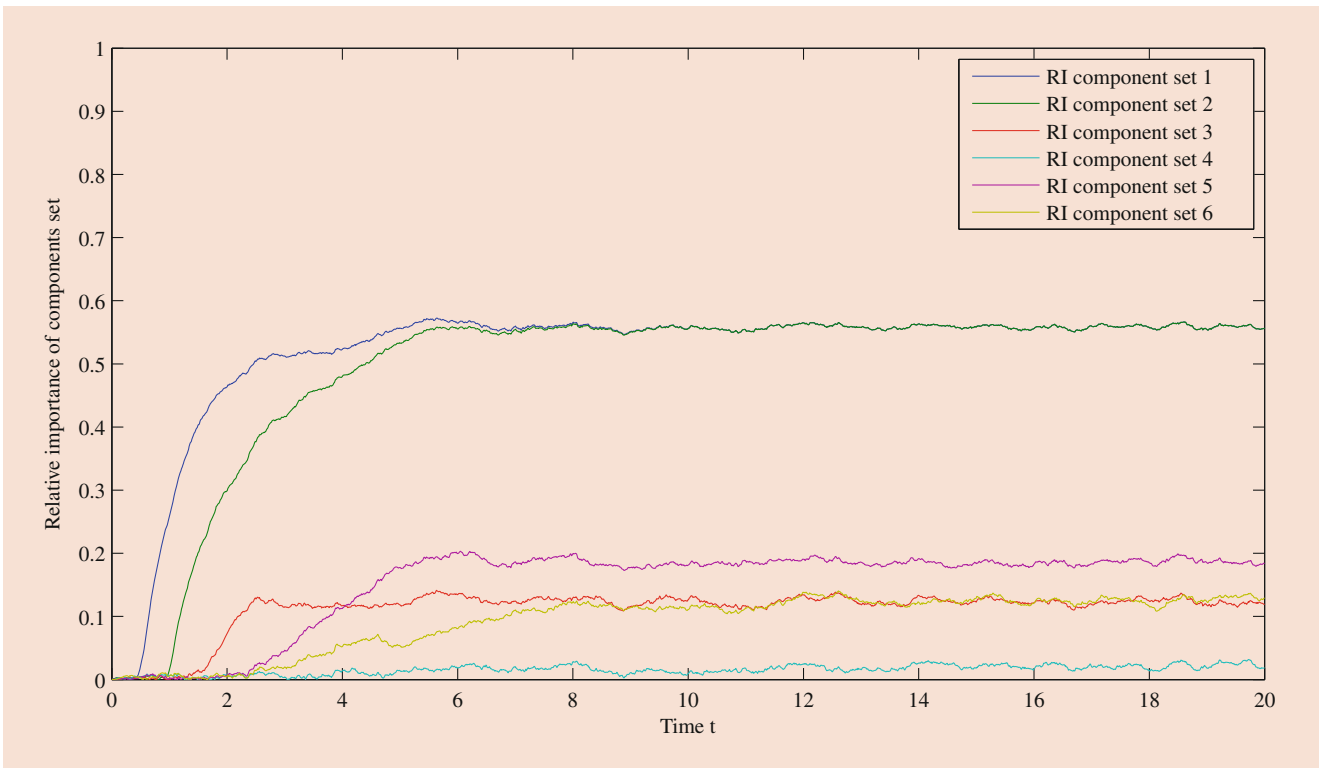


Fig. 21.6 Relative importance index of the component sets with same type in system

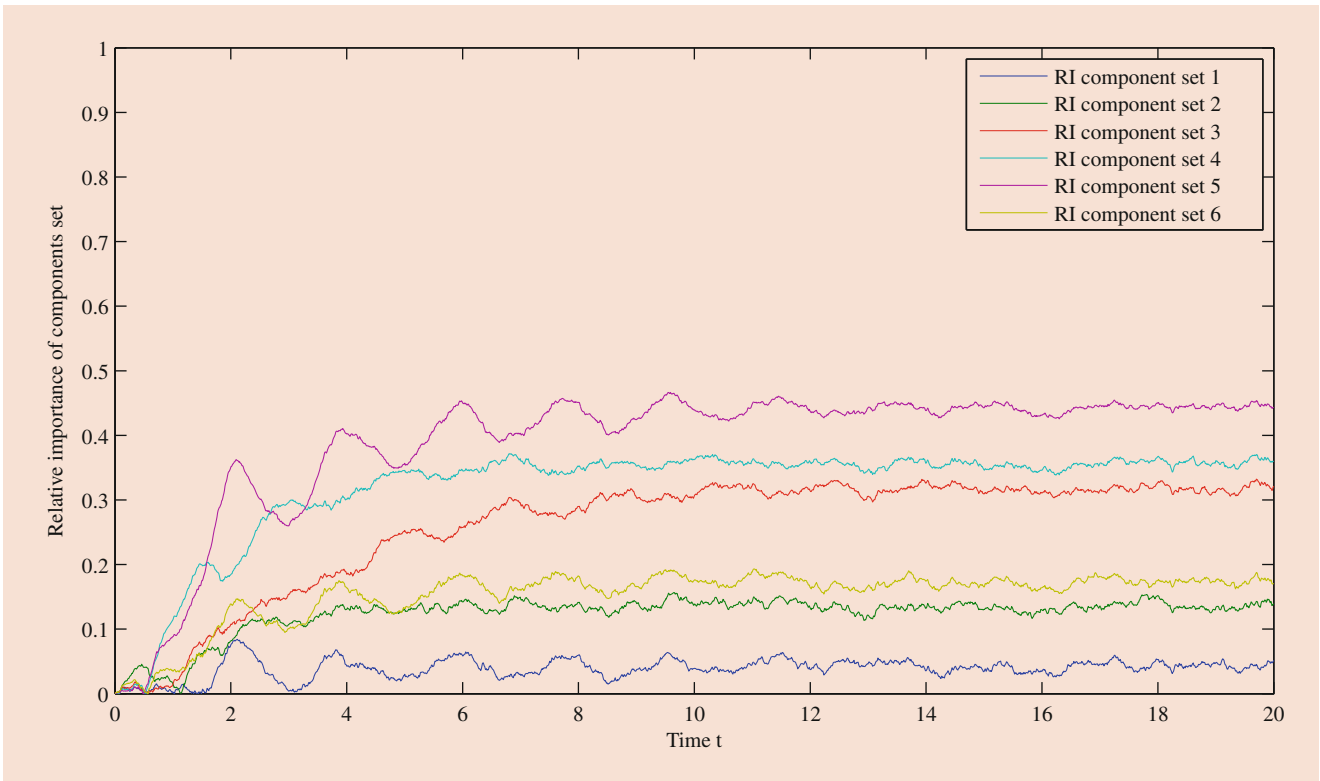


Fig. 21.7 Relative importance index of the component sets with different types in system

sets, that is, set 1 with three components (5, 7, 9), set 2 with components (1, 6, 10), set 3 with components (3, 4, 13), set 4 with components (2, 8, 13), set 5 with components (12, 14), and set 6 with four components (1, 3, 7, 9). Figure 21.7 indicates the importance degree of these six component sets which belong to different component types.

It can be gotten from the above figure that the relative importance index value of different sets at each time. For example, at time $t = 8$, component set 5 has the biggest influence on this repairable system. Then it comes to component set 4 and 3. The component sets 6 and 2 ranked the fourth and fifth, respectively, while component set 1 has the least relative importance index value.

If using the quantitative importance index to quantify the importance degree during the survival time, the QI of a specific component and different types component set can be seen in Figs. 21.8 and 21.9, respectively.

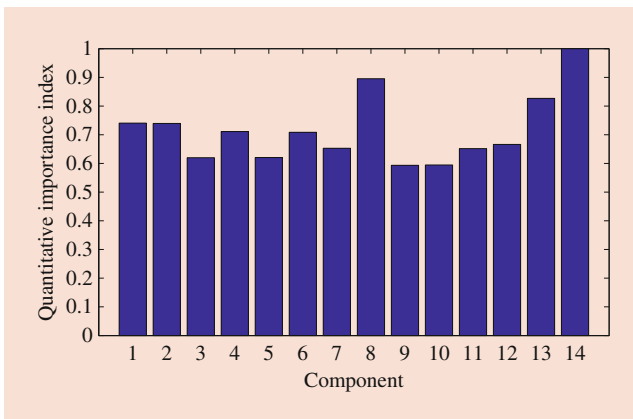


Fig. 21.8 Quantitative importance index of the specific component in the system

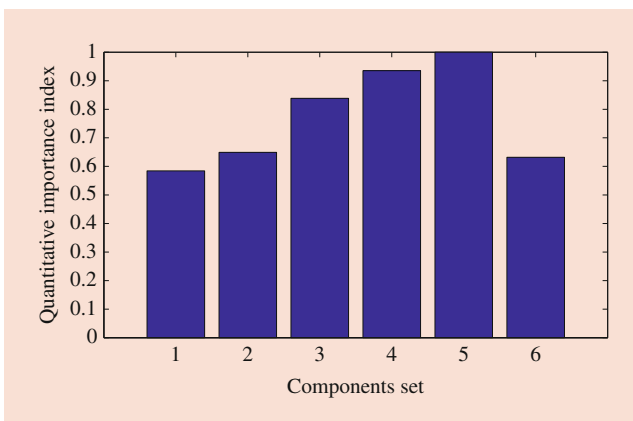


Fig. 21.9 Quantitative importance index of the component sets with different types in the system

The first figure shows that component 14 is the most “critical” one to the whole system, while the second figure indicates set 5 with components (12, 14) has the biggest influence degree on the repairable system.

21.5.2 Imprecision Within the System

Due to lack to exact data or the confidential contract of the company, the precise configuration of the system or the survival signature cannot be gotten. Figure 21.10 shows a coherent system with an imprecise part, that is, the configuration of the fifteen components which belong to five component types cannot be known exactly. To be specific, the unknown part of the system is composed of two components of type 5 and one component of type 3 and 6, respectively. Therefore, the survival signature of this coherent system is imprecise with bounds. What is more, there exists imprecision within the component failure times, just as Table 21.2 indicates.

The upper and lower survival function bounds of the coherent imprecise system can be calculated through the Monte Carlo simulation method, which is performed by using 100,000 samples. The reliability intervals of the imprecise system can be seen in Fig. 21.11. This example indicates that the simulation method is general to deal with imprecise systems, despite the imprecision within the survival signature or component failure parameters.

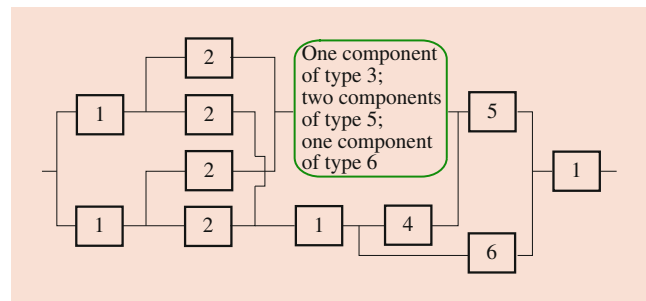


Fig. 21.10 Imprecise coherent system with fifteen components which belong to five types

Table 21.2 Distribution for failure and repair process of components

Component type	Distribution type	Imprecise parameters
1	Exponential	[2.1, 2.5]
2	Exponential	[0.9, 1.4]
3	Weibull	[(1.6, 1.8], [3.3, 3.9])
4	Lognormal	[(1.3, 1.8], [2.3, 2.9])
5	Weibull	[(3.1, 3.3], [2.3, 2.7])
6	Gamma	[(2.9, 3.3], [1.3, 1.8])

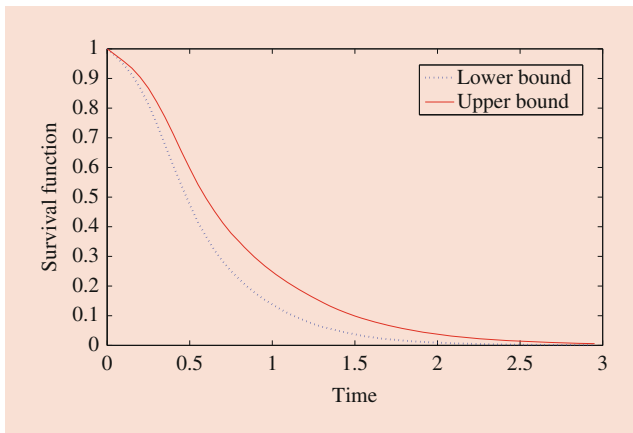


Fig. 21.11 Intervals of the survival function of the imprecise system in Fig. 21.10

21.6 Conclusions

This paper mainly focuses on complex repairable systems. The efficient simulation methods which are based on structure function and survival signature have been proposed, respectively. The survival signature is a summary of structure function, which opens a new pathway for reliability analysis on complex repairable systems.

In order to find out which component or component set is “critical” to the whole repairable system, the importance and sensitivity analysis is needed. Therefore, engineers can allocate the limited resources to the most important component or component set. To be specific, relative importance index of a specific component and a set of components which based on structure function and survival signature are presented in this paper. All of relative importance index are time-dependent, so it is easy to identify which component has the biggest influence degree on the repairable system at each time. In order to quantify importance degree of them during the survival time, the quantitative importance index is introduced.

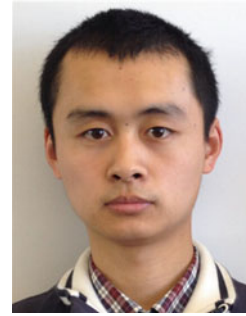
Sometimes the full configuration of the system is unknown if there has confidential contract or only know the bounds of survival signature due to lack of data. What is more, there exist imprecision within the component failure parameters. All of these will lead to survival function intervals of the system.

The feasibility and effectiveness of the proposed measures which based on simulation method are demonstrated by some numerical examples; the results show that the approaches are useful for analyzing component importance of complex repairable systems and the system with imprecision.

References

- Aslett, L.J.: Reliability theory: tools for structural reliability analysis. r package (2012). www.louisaslett.com
- Beer, M., Ferson, S., Kreinovich, V.: Imprecise probabilities in engineering analyses. *Mech. Syst. Signal Process.* **37**(1), 4–29 (2013)
- Birnbaum, Z.W.: On the importance of different components in a multicomponent system. Technical report. DTIC Document (1968)
- Borgonovo, E.: A new uncertainty importance measure. *Reliab. Eng. Syst. Saf.* **92**(6), 771–784 (2007)
- Borgonovo, E., Apostolakis, G.E.: A new importance measure for risk-informed decision making. *Reliab. Eng. Syst. Saf.* **72**(2), 193–212 (2001)
- Coolen, F.: On the use of imprecise probabilities in reliability. *Qual. Reliab. Eng. Int.* **20**(3), 193–202 (2004)
- Coolen, F.P., Coolen-Maturi, T.: Generalizing the signature to systems with multiple types of components. In: *Complex Systems and Dependability*, pp. 115–130. Springer, Berlin (2012)
- Coolen, F.P., Coolen-Maturi, T.: Modelling uncertain aspects of system dependability with survival signatures. In: *Dependability Problems of Complex Information Systems*, pp. 19–34. Springer, Berlin (2015)
- Coolen, F.P., Coolen-Maturi, T.: The structure function for system reliability as predictive (imprecise) probability. *Reliab. Eng. Syst. Saf.* **154**, 180–187 (2016)
- Coolen, F.P., Troffaes, M.C., Augustin, T.: Imprecise probability. In: *International Encyclopedia of Statistical Science*, pp. 645–648. Springer, Berlin (2011)
- Coolen, F.P., Coolen-Maturi, T., Al-Nefaiee, A.H.: Nonparametric predictive inference for system reliability using the survival signature. *Proc. Inst. Mech. Eng. O J. Risk Reliab.* **228**(5), 437–448 (2014)
- Ertas, A., Smith, M.W., Tate, D., Lawson, W.D., Baturalp, T.B.: Complexity of system maintainability analysis based on the interpretive structural modeling methodology: transdisciplinary approach. *J. Syst. Sci. Syst. Eng.* **25**(2), 254–268 (2016)
- Feng, G.: Survival signature-based reliability approach for complex systems susceptible to common cause failures. *Int. J. Reliab. Qual. Saf. Eng.* **26**(03), 1950013 (2019)
- Feng, G., Patelli, E., Beer, M., Coolen, F.P.: Imprecise system reliability and component importance based on survival signature. *Reliab. Eng. Syst. Saf.* **150**, 116–125 (2016)
- Ferson, S., Hajagos, J.G.: Arithmetic with uncertain numbers: rigorous and (often) best possible answers. *Reliab. Eng. Syst. Saf.* **85**(1), 135–152 (2004)
- Ferson, S., Kreinovich, V., Ginzburg, L., Myers, D.S., Sentz, K.: *Constructing Probability Boxes and Dempster-Shafer Structures*, vol. 835. Sandia National Laboratories, Albuquerque (2002)
- Fussell, J.: How to hand-calculate system reliability and safety characteristics. *IEEE Trans. Reliab.* **3**, 169–174 (1975)
- George-Williams, H., Feng, G., Coolen, F.P., Beer, M., Patelli, E.: Extending the survival signature paradigm to complex systems with non-repairable dependent failures. *Proc. Inst. Mech. Eng. O J. Risk Reliab.* 1748006X18808085 (2018)
- Levitin, G., Lisnianski, A.: Importance and sensitivity analysis of multi-state systems using the universal generating function method. *Reliab. Eng. Syst. Saf.* **65**(3), 271–282 (1999)
- Modarres, M.: *What Every Engineer Should Know About Reliability and Risk Analysis*, vol. 30. CRC Press, Boca Raton (1992)

21. Modarres, M.: Risk Analysis in Engineering: Techniques, Tools, and Trends. CRC Press, Boca Raton (2006)
22. Patelli, E., Feng, G., Coolen, F.P., Coolen-Maturi, T.: Simulation methods for system reliability using the survival signature. *Reliab. Eng. Syst. Saf.* **167**, 327–337 (2017)
23. Pham, H.: Handbook of Reliability Engineering. Springer Science & Business Media, Berlin (2006)
24. Reed, S.: An efficient algorithm for exact computation of system and survival signatures using binary decision diagrams. *Reliab. Eng. Syst. Saf.* **165**, 257–267 (2017)
25. Samaniego, F.J.: System Signatures and Their Applications in Engineering Reliability, vol. 110. Springer Science & Business Media, Berlin (2007)
26. Song, K.Y., Chang, I.H., Pham, H.: A three-parameter fault-detection software reliability model with the uncertainty of operating environments. *J. Syst. Sci. Syst. Eng.* **26**(1), 121–132 (2017)
27. Vesely, W.: A time-dependent methodology for fault tree evaluation. *Nucl. Eng. Des.* **13**(2), 337–360 (1970)
28. Wang, W., Loman, J., Vassiliou, P.: Reliability importance of components in a complex system. In: Reliability and Maintainability, 2004 Annual Symposium-RAMS, pp. 6–11. IEEE, Piscataway (2004)
29. Zio, E., Baraldi, P., Patelli, E.: Assessment of the availability of an offshore installation by monte carlo simulation. *Int. J. Press. Vessel. Pip.* **83**(4), 312–320 (2006)






Dr. Geng Feng joined in the School of Engineering, University of Central Lancashire as a Lecturer in 2018. He has been working as a researcher in University of Bristol and Thales UK, Ltd. after he obtained his Ph.D. degree from University of Liverpool in 2017. Dr Feng's main research interests include (1) RAMS (reliability, availability, maintainability, and safety) of complex systems; (2) imprecision theory and applied mathematics; (3) sensitivity analysis and component importance measures; (4) design for reliability and disaster mitigation.



Hardware and Software Reliability, Verification, and Testing

22

Ashis Kumar Chakraborty, E. V. Gijo , Anisha Das , and Moutushi Chatterjee 

Contents

22.1	Hardware Reliability	415
22.1.1	Introduction	416
22.1.2	Reliability Models/Distributions	417
22.1.3	System Reliability	419
22.1.4	Censoring	421
22.1.5	Accelerated Life Testing	423
22.1.6	Repairable System Reliability	424
22.1.7	Verification and Validation	425
22.2	Software Reliability	426
22.2.1	Introduction	426
22.2.2	Software Reliability Models	429
22.2.3	Software Reliability Growth Models	430
22.2.4	Fault Count (FC) Models	431
22.2.5	Fault Seeding and Input Domain based Models	433
22.3	Optimum Release Time Models	434
22.4	Software Testing and Validation	435
22.4.1	Introduction to Software Testing	435
22.4.2	Some Basic Definitions	435
22.4.3	What Is Software Testing?	435
22.4.4	The Psychology of Testing	436
22.4.5	Who Does Testing?	436
22.4.6	The Optimal Time for Testing	437
22.4.7	Black-Box and White-Box Testing	437
22.4.8	To Conclude Testing	437
22.4.9	Software Verification and Validation	437
22.4.10	Software Validation as Part of System Design	438
22.4.11	Principles of Software Validation	438
22.4.12	Benefits of Software Validation	440
22.4.13	Validation vs. Verification	440
22.4.14	Conclusion	440
References		440

Abstract

Hardware and software together are now part and parcel of almost all the modern devices. Hence, the study of both hardware reliability and software reliability has become very important in order to ensure availability of the devices. There are several distinct differences between hardware and software, and hence, even though the definition of reliability in both the cases remains the same, finding out reliability of a hardware may call for a different methodology than that for a software. Since a software cannot be seen, nor can it be touched, finding out reliability of a software becomes difficult as such. In this chapter we discuss in brief the concepts and methodologies adopted to find out reliabilities of software and hardware. We also discuss some basic differences between hardware and software. A few important methods used for estimating hardware and software reliability have been discussed in brief. A thorough bibliography has been provided for the readers to look into the details of the methodologies wherever required.

Keywords

Hardware and software reliability · Hazard rate · System reliability · Censoring · Accelerated life testing · Repairable system · Maintainability · Software validation · Software testing · Optimum time for testing

22.1 Hardware Reliability

This chapter discusses about various reliability concepts. It starts with a general introduction to reliability data and usefulness of reliability analysis and modeling. Further, various types of reliability modeling including system reliability, censoring, accelerated life testing, repairable system, etc. are discussed. This chapter also presents a list of references.

A. K. Chakraborty (✉) · A. Das
 Indian Statistical Institute, Kolkata, West Bengal, India
 E. V. Gijo
 Indian Statistical Institute, Bangalore, Karnataka, India
 M. Chatterjee
 Lady Brabourne College, Kolkata, West Bengal, India

22.1.1 Introduction

The reliability engineering field was developed mainly during World War II. Problems related to maintenance and availability of equipment forced the Department of Defense to develop the area of reliability engineering. Initially, Department of Defense, through sponsored research, developed a significant part of reliability theory. Federal Aviation Administration and Department of Defense recognized the urgent need to develop Reliability requirements with the introduction of solid-state electronics and more complex equipments. Reliability is no longer a desirable, but vague, adjective; it is now a well-defined and often specifically required statistic. It has also gained momentum among practicing engineers and researchers. Reliability concepts in all phases of the product life cycle from design to manufacture have resulted in developing cost-effective systems that result in better performance. In the medical field, while studying about the life of patients, survival analysis is widely used.

Earlier, the product designers were primarily responsible for creating a design that would meet the “performance” requirements. For example, a pump should do the needed pumping at good efficiency. Now there is a paradigm shift in this thought process. Designers are strongly influenced by the fact that the “performance” requirements need to be sustainable for a specified period of time.

A general definition of reliability can be the probability that a product will perform the specified functions without failure, under required environmental and operational conditions, for a defined period of time. This definition focuses on four important points: probability of survival, performance requirements of the product, customer usage conditions, and expected life or time to failure [33].

Reliability Function

Let the random variable X be the lifetime or the time to failure of a component. The probability that the component survives until sometime t is called the reliability $R(t)$ of the component:

$$R(t) = P(X > t) = 1 - F(t) \tag{22.1}$$

where F is the distribution function of the component lifetime, X . Thus:

$$R(t) = P(X > t) = \int_t^\infty f(x)dx \tag{22.2}$$

$f(x)$ is the density function.

If the random variable X follows exponential distribution with failure rate λ , then the reliability function $R(t)$ can be estimated as:

$$R(t) = \int_t^\infty \lambda e^{-\lambda x} dx \tag{22.3}$$

Hence, in this case, $R(t) = e^{-\lambda t}$. Thus, the reliability function can be estimated for the failure data of system, sub-system, or component after identifying the underlying distribution.

Expected Life

The expected life or the expected time during which a component will perform successfully, is defined as:

$$E(t) = \int_0^\infty xf(x)dx \tag{22.4}$$

When the system being tested is renewed through maintenance and repairs, $E(t)$ is also known as the mean time between failures (MTBF) otherwise mean time to failure (MTTF).

The Failure Rate and Hazard Rate Function

The probability of failure of a system in a given time interval $[t_1, t_2]$ can be expressed in terms of the reliability function as:

$$\int_{t_1}^{t_2} f(t)dt = \int_{t_1}^\infty f(t)dt - \int_{t_2}^\infty f(t)dt = R(t_1) - R(t_2) \tag{22.5}$$

The rate at which the failure occurs in a certain interval of time $[t_1, t_2]$ is called the failure rate during that interval. It is defined as the conditional probability that a failure occurs in the interval, given that a failure has not occurred prior to t_1 , the beginning of the interval. Thus, the failure rate is:

$$\frac{R(t_1) - R(t_2)}{(t_1 - t_2)R(t_1)} \tag{22.6}$$

If the interval is refined as $[t, t + \Delta t]$, then the above equation becomes:

$$\frac{R(t) - R(t, t + \Delta t)}{\Delta t R(t)} \tag{22.7}$$

The hazard function is defined as the limit of the failure rate as the interval approaches zero. Thus, the hazard function is the instantaneous failure rate. The hazard function $h(t)$ is defined as:

$$h(t) = \lim_{\Delta t \rightarrow 0} \frac{R(t) - R(t, t + \Delta t)}{\Delta t R(t)} = \frac{1}{R(t)} \left[-\frac{d}{dt} R(t) \right] = \frac{f(t)}{R(t)} \tag{22.8}$$

The Concept of the Bathtub Curve

The hazard function $h(t)$ will change over the lifetime of products. The whole pattern of failures could be depicted

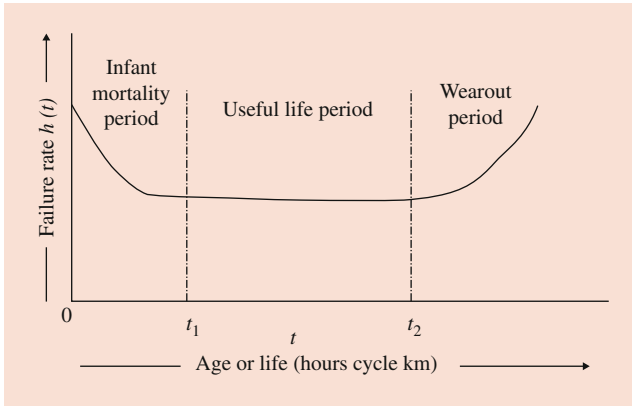


Fig. 22.1 Bathtub curve

by a bathtub curve as shown in Fig. 22.1. This curve can be divided into three different parts, viz., infant mortality period, useful life period, and wear-out period (refer Nelson [48]).

Infant Mortality Period

Infant mortality period is characterized by rapidly decreasing failure rate starting with high failure rate, the failures show up in the form of poor welds or seals, poor solder joints, poor connections, contamination on surfaces, chemical impurities in metals, voids/cracks in protective coatings, incorrect positioning of parts, etc. Early failures can be prevented through improved process control during manufacturing. The practice of shaking-off and burning-in on new products and equipment is designed to discover the weak items and get rid of them before further usage or dispatch to customers.

Useful Life Period

Useful life period is characterized by chance failure, which means, any failure whose cause and/or mechanism makes its time of occurrence unpredictable but which is predictable only in a probabilistic or statistical sense. When the lifetime is governed by exponential law, the failure rate during useful life period will be a constant, denoted by, λ . The causes of failure are the limitations inherent in design, usage, and maintenance. The prevention of failures can be achieved through:

- Improvement in design
- Provision for redundancy
- Better maintenance

Wear-Out Period

This period is characterized by rapidly increasing failure rate with increasing age as a result of deterioration processes or mechanical wear. The deterioration process may include corrosion/oxidation, insulation breakdown or leakage, ionic

migration of metals in vacuum or on surfaces, and shrinkage or cracks in plastics. When the failure rate due to wear out becomes unacceptably high, replacement or repair of the item is to be undertaken.

22.1.2 Reliability Models/Distributions

Quite a few distributions are used for modelling the life data. A few of the most commonly used distributions and the corresponding reliability functions are discussed below.

Exponential Distribution

Exponential distribution is widely used in reliability modelling and prediction because of its simple properties. An exponential distribution with failure rate λ is defined as:

$$f(t) = \lambda e^{-\lambda t}, \quad t \geq 0 \text{ and } \lambda > 0 \quad (22.9)$$

The reliability function:

$$R(t) = e^{-\lambda t}, \quad t \geq 0 \quad (22.10)$$

The hazard function is:

$$h(t) = \frac{f(t)}{R(t)} = \lambda \quad (22.11)$$

Log Normal Distribution

The density function of log normal distribution is

$$f(t) = \frac{1}{\rho t \sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{\ln t - \mu}{\sigma} \right)^2 \right], \quad t \geq 0 \quad (22.12)$$

The cumulative distribution function:

$$F(t) = \int_0^t \frac{1}{\rho x \sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{\ln x - \mu}{\sigma} \right)^2 \right] dx \quad (22.13)$$

This can be written in terms of the standard normal deviate z as:

$$F(t) = P[X \leq t] = P \left[Z \leq \frac{\ln t - \mu}{\sigma} \right]$$

The reliability function is:

$$R(t) = P[X > t] = P \left[Z > \frac{\ln t - \mu}{\sigma} \right] \quad (22.14)$$

The hazard function is:

$$h(t) = \frac{f(t)}{R(t)} = \frac{\phi \left(\frac{\ln t - \mu}{\sigma} \right)}{t \sigma R(t)} \quad (22.15)$$

where ϕ is the standard normal probability density function and μ and σ are the mean and standard deviation of the natural logarithm of time to failure.

Weibull Distribution

The probability density function for Weibull distribution is:

$$f(t) = \frac{\beta(t-\delta)^{\beta-1}}{(\theta-\delta)^\beta} \exp\left[-\left(\frac{t-\delta}{\theta-\delta}\right)^\beta\right], t \geq \delta \geq 0 \quad (22.16)$$

where β is the shape parameter and $(\theta - \delta)$ is the scale parameter. The reliability and hazard function are, respectively:

$$R(t) = \exp\left[-\left(\frac{t-\delta}{\theta-\delta}\right)^\beta\right] \quad (22.17)$$

$$h(t) = \frac{\beta(t-\delta)^{\beta-1}}{(\theta-\delta)^\beta} \quad (22.18)$$

Gamma Distribution

The density function, reliability function, and the hazard function for gamma distribution are as follows:

$$f(t) = \frac{\lambda^\eta}{\Gamma(\eta)} t^{\eta-1} e^{-\lambda t}, t \geq 0, \eta > 0, \lambda > 0, \quad (22.19)$$

where η is the shape parameter and λ is the scale parameter.

$$R(t) = 1 - F(t) = \sum_{k=0}^{\eta-1} \frac{(\lambda t)^k \exp[-\lambda t]}{k!} \quad (22.20)$$

$$h(t) = \frac{\frac{\lambda^\eta}{\Gamma(\eta)} t^{\eta-1} e^{-\lambda t}}{\sum_{k=0}^{\eta-1} \frac{(\lambda t)^k \exp[-\lambda t]}{k!}} \quad (22.21)$$

Smallest Extreme Value Distribution

The density function, reliability function, and the hazard function for the **smallest extreme value distribution** are as follows:

$$f(t) = \frac{1}{\sigma} \Phi_{SEV} \left(\frac{t-\mu}{\sigma} \right) \quad (22.22)$$

Where $-\infty < \mu < \infty$ is the location parameter, $\sigma > 0$ is the scale parameter, and:

$$\Phi_{SEV} = \exp\left(\frac{t-\mu}{\sigma}\right) \exp\left\{-\exp\left(\frac{t-\mu}{\sigma}\right)\right\} \quad (22.23)$$

$$R(t) = 1 - \Phi_{SEV} \left(\frac{t-\mu}{\sigma} \right) \quad (22.24)$$

$$h(t) = \frac{1}{\sigma} \exp\left(\frac{t-\mu}{\sigma}\right) \quad (22.25)$$

Largest Extreme Value Distribution

The density function, reliability function, and the hazard function for the **largest extreme value distribution** are as follows:

$$f(t) = \frac{1}{\sigma} \Phi_{LEV} \left(\frac{t-\mu}{\sigma} \right) \quad (22.26)$$

where $-\infty < \mu < \infty$ is the location parameter, $\sigma > 0$ is the scale parameter, $-\infty < t < \infty$ and $\Phi_{LEV} = \exp\left(\frac{\mu-t}{\sigma}\right) \exp\left\{-\exp\left(\frac{\mu-t}{\sigma}\right)\right\}$

$$R(t) = 1 - \Phi_{LEV} \left(\frac{t-\mu}{\sigma} \right) \quad (22.27)$$

$$h(t) = \frac{\exp\left(\frac{t-\mu}{\sigma}\right)}{\sigma \left\{ \exp\left[\exp\left(\frac{t-\mu}{\sigma}\right)\right] - 1 \right\}} \quad (22.28)$$

Logistic Distribution

The density function, reliability function, and the hazard function for **logistic distribution** are as follows:

$$f(t) = \frac{1}{\sigma} \Phi_{logis} \left(\frac{t-\mu}{\sigma} \right) \quad (22.29)$$

where $-\infty < \mu < \infty$ is the location parameter and $\sigma > 0$ is the scale parameter.

$$R(t) = 1 - \Phi_{logis} \left(\frac{t-\mu}{\sigma} \right) \quad (22.30)$$

$$h(t) = \frac{1}{\sigma} \Phi_{logis} \left(\frac{t-\mu}{\sigma} \right) \quad (22.31)$$

Loglogistic Distribution

The density function, reliability function, and the hazard function for **log-logistic distribution** are as follows:

$$f(t) = \frac{1}{\sigma t} \Phi_{logis} \left(\frac{\log t - \mu}{\sigma} \right) \quad (22.32)$$

where $\sigma > 0$ is the scale parameter, $t > 0$, and:

$$\Phi_{logis} = \frac{\exp\left(-\frac{\ln t - \mu}{\sigma}\right)}{\left[1 + \exp\left(-\frac{\ln t - \mu}{\sigma}\right)\right]^2}$$

$$R(t) = 1 - \Phi_{logis} \left(\frac{\log t - \mu}{\sigma} \right) \quad (22.33)$$

$$h(t) = \frac{1}{\sigma t} \Phi_{logis} \left(\frac{\log t - \mu}{\sigma} \right) \quad (22.34)$$

22.1.3 System Reliability

A system consisting of two or more parts or components is designed to perform one or more of functions. In different systems, these components can have different configurations. The reliability-wise configuration of the components to be studied to prepare a reliability block diagram. The estimation and prediction of system reliability is based on the type of configuration of the components or subsystems. These configurations can be a pure series or parallel arrangement or a very complex configuration. The following are the most commonly used system configurations.

- Series system
- Parallel system
- k-out-of-n system
- Complex system

Series System

The series system is the one in which every component must function if the system is to function. Complex systems are sometimes subdivided into a series arrangement for analysis by properly grouping components into a unit such that the units are in series. A schematic representation of the series system is provided in Fig. 22.2.

Let t_i be the random variable of the time to failure for the i th component; then for an n component series system, the system reliability is:

$$R_S(t) = P[t_1 > t \cap t_2 > t \cap \dots \cap t_n > t] \quad (22.35)$$

If we assume independence, then:

$$R_S(t) = P(t_1 > t)P(t_2 > t) \dots P(t_n > t)$$

$$\text{or, } R_S(t) = \prod_{i=1}^n P(t_i > t) = \prod_{i=1}^n R_i(t) \quad (22.36)$$

Example 1 A simple computer consists of a processor, a bus, and a memory. The computer will work only if all three are functioning correctly (Fig. 22.3). The probability that the processor is functioning is 0.99, that the bus is functioning 0.95, and that the memory is functioning is 0.99. What is the probability that the computer will work?

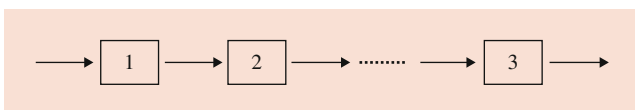


Fig. 22.2 Reliability block diagram for series system

Solution: System Reliability = $0.99 \times 0.95 \times 0.99 = 0.931095$. So, even though all the components have 95% or more reliability, the overall system reliability of the system is only 93%.

Example 2 An electronic product contains 100 integrated circuits. The probability that any integrated circuit is defective is 0.001, and the integrated circuits are independent. The product operates only if all the integrated circuits are operational. What is the probability that the product is operational?

Solution: The probability that any component is functioning is 0.999. Since the product operates only if all 100 components are operational, the probability that the 100 components are functioning is system reliability = $(0.999)^{100} = 0.9047921$. The reliability is just over 90% even though each component has a reliability of 99.9%.

Parallel System

This is a system that will fail only if all the components or subsystems fail. This means any of the elements in parallel structure permit the system function. A block diagram of the parallel system is presented in Fig. 22.4.

In this case, the system will work if at least one of the n components works. Hence, the expression for system reliability is:

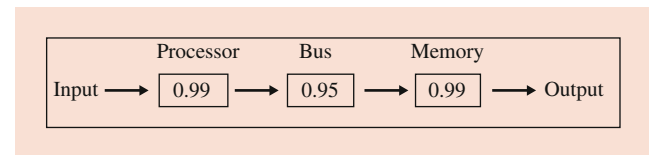


Fig. 22.3 Reliability block diagram for Example 1

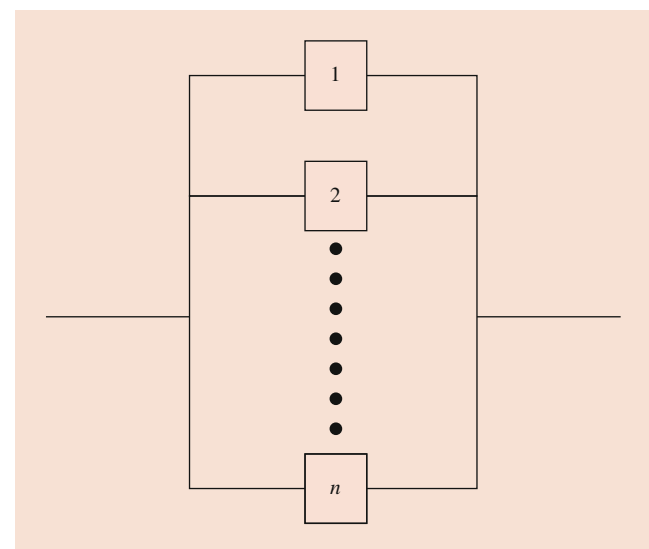


Fig. 22.4 Reliability block diagram for parallel system

$$R_S = 1 - (1 - R_1)(1 - R_2) \cdots (1 - R_n) = 1 - \prod_{i=1}^n (1 - R_i) \tag{22.37}$$

Example 3 A subassembly has a high failure rate of $\lambda = 0.002$ per hour. As an insurance backup, a second subassembly is kept in a standby mode. What is the reliability of such a system for a period of thousand hours?

Solution: Here, $R = R_1 = R_2 = e^{-0.002 \times 1000} = 0.8187$, $R_S = 1 - (1 - R)(1 - R) = 0.9671$.

K-out-of-n-structure

A k-out-of-n structure is defined as a system which works, if and only if, at least k components out of the n the components work where n is the total number of components and $1 \leq k \leq n$. The k-out-of-n system is one of the most commonly used configurations. In this configuration, all the components have the same failure distribution, and the failure of one component is independent of the failure of other components.

The reliability of a k-out-of-n system can be estimated with the help of binomial distribution. In this case, the system reliability R_S can be defined as:

$$R_S = \sum_{r=k}^n \binom{n}{r} R^r (1 - R)^{n-r} \tag{22.38}$$

where n is the number of units in the system and k is the minimum number of units required for functioning of the system and R is the reliability of each unit in the system.

Example 4 A system consists of six components of which at least four must function properly for the system to perform. Each component has a reliability of 0.85 during the functioning of the system. Estimate the probability of successful performance of the system.

Solution: This system can be considered as a k-out-of-n system, where $k = 4$ and $n = 6$,

$$R_S = \sum_{r=4}^6 0.85^r (.85)^{6-r} = 95.26\%$$

The effect of increasing the number of units required for successful performance of the system can be studied while the total number of units remains constant (in this example, six units). The Table 22.1 provides, the reliability of the k-out-of-6 configuration versus different numbers of required units.

Complex Systems

Certain design configurations may produce systems in which pure parallel or series configurations are not appropriate. Most of the smaller systems can be represented either by a

Table 22.1 Reliability for a k-out-6 system for different k values

k	Reliability
1	0.9999
2	0.9996
3	0.9941
4	0.9526
5	0.7765
6	0.3772

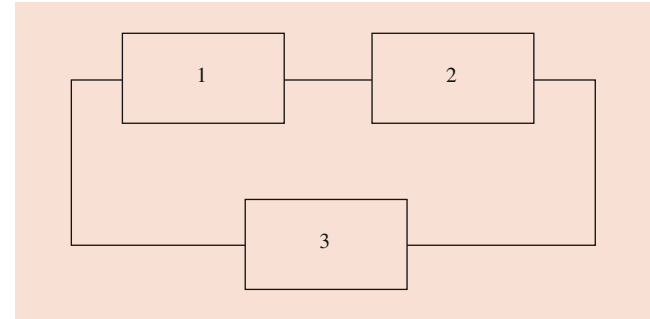


Fig. 22.5 Reliability block diagram for Example 5

simple series or parallel configuration. But, there are large systems, which may be a combination of series and parallel configurations. The reliability of such systems can be estimated by considering the individual series or parallel subsystems. These subsystems can be further combined to get the reliability estimate.

Example 5 Suppose a complex system consists of three units/subsystems in which subsystems 1 and 2 are connected in series. Further, subsystem 3 is connected in parallel with the first two subsystems, as shown in Fig. 22.5. Calculate the system reliability at 100 hours if $R_1 = 0.995$, $R_2 = 0.987$, and $R_3 = 0.973$. First, the reliability of the series segment consisting of Units 1 and 2 is calculated as, $R_{1,2} = R_1 \times R_2 = 0.995 \times 0.987 = 0.9820654$. The reliability of the overall system is then calculated by treating Units 1 and 2 as one unit with a reliability of 98.2065% connected in parallel with Unit 3. Therefore, the system reliability R_S is:

$$\begin{aligned} R_S &= 1 - (1 - 0.982065)(1 - 0.973) \\ &= 1 - 0.17935 \times 0.027 \\ &= 0.99515755 \\ &= 99.95\% \end{aligned}$$

Example 6 Consider an equipment/system which has seven subsystems with the following configuration (Fig. 22.6). Estimate the system reliability by considering the subsystem reliability values provided in the diagram.

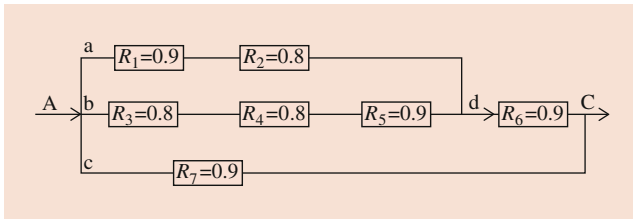


Fig. 22.6 Reliability block diagram for Example 6

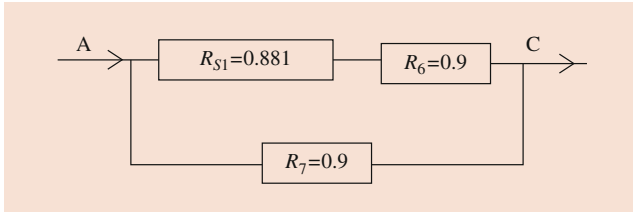


Fig. 22.7 Modified reliability block diagram for Example 6

For calculating the system reliability, first, subsystems which are connected as series or parallel can be considered. Further, by combining the subsystems, the system reliability can be calculated. For example:

$$R_a = R_1 R_2 = 0.72$$

$$R_b = R_3 R_4 R_5 = 0.576$$

Now, R_a and R_b are connected in parallel, and hence its reliability can be estimated as $R_{S1} = 0.881$. Now the system can be represented as presented in Fig. 22.7.

Using similar arguments, the system reliability $R_S = 0.979$. Thus, the reliability of the system is 0.979.

22.1.4 Censoring

Consider a situation of performing reliability testing of n (non-repairable) units taken randomly from a population. We are investigating the population to determine if its failure rate is acceptable.

A life testing or observing the life data for a product/component in real life can be censored at a particular time or when a particular number of failures are observed. In fact the test can continue till all the products/components fail. But this will prolong the test duration and become very expensive. Also sometimes it is not possible to test till all the units in test fail. Hence, most of the life tests are censored. There are various types of censoring schemes available. The most common among them are known as type-1 censoring, type-2 censoring, hybrid censoring, and interval censoring.

Type I Censoring

Consider a life testing scenario with n units under test. This test is continued until time period T and observe how many of these units have failed and how many of them are surviving. This type of censoring scheme is known as *Type I censoring*. During this time period T , let the number of failures be r , $0 \leq r \leq n$. Let t_1, t_2, \dots, t_r be the exact time to failure of these r units. The remaining $n-r$ units survived after the time period T . In this case T is fixed and r is random. The data on time to failure for the r failed units are recorded. This censoring scheme is also known as “right censoring,” as data larger than T is not available. Let, in this case, the mean life $\hat{\theta}$ can be estimated as follows:

Define δ_i , such that $\delta_i = 1$, if the unit has failed and $\delta_i = 0$, if the unit is censored.

$$\hat{\theta} = \frac{\sum_{i=1}^n \{\delta_i t_i + (1 - \delta_i) T\}}{r} \tag{22.39}$$

where “ n ” is the number of units in test, “ T ” is the time interval, and “ r ” is the number of failures. The confidence interval for $\hat{\theta}$ can be obtained as:

$$X_{1-\frac{\alpha}{2}, 2r}^2 \leq \frac{2r\hat{\theta}}{\theta} \leq X_{\frac{\alpha}{2}, 2r}^2 \tag{22.40}$$

Type II Censoring

One disadvantage of Type-1 censoring scheme is that, there may not be any failures at all during the time period T , under observation. Hence, one can perform a life test until r failures are observed out of n units under study. In this case, the test is continued till r units have failed. As an example, consider a life test with 50 units and the reliability engineer would like to continue the test till 25 of them fails. In this case $n = 50$ and $r = 25$. Also, note that the total time of the life test is unknown until the test is completed. This type of censoring scheme is known as Type-II censoring. In this case, the observed failure times are t_1, t_2, \dots, t_r , where r is fixed prior to the test. In this case also, $n - r$ units have survived after the termination of the test.

In this scheme, it is known in advance regarding the total number of failed units during the test. But, the termination time of the test is not known in advance.

Type-II censoring scheme is also known as failure censoring, as the test is terminated after a definite number of failures are observed.

The mean life θ can be estimated as:

$$\hat{\theta} = \frac{\sum_{i=1}^r X_i + (n - r)X_r}{r} \tag{22.41}$$

where “ n ” is the number of units in test and “ r ” is the number of failures. The confidence interval for “ θ ” is defined as:

$$X^2_{1-\frac{\alpha}{2}, 2r} \leq \frac{2r\hat{\theta}}{\theta} \leq X^2_{\frac{\alpha}{2}, 2(r+1)} \tag{22.42}$$

Hybrid Censoring

The hybrid censoring scheme is a mixture of type-I and type-II censoring schemes. The test is terminated when a pre-fixed number, $r < n$, out of n items have failed, or when a pre-fixed time T , has been reached. In other words, the life test is terminated at a random time $T^* = \min(t_r, T)$, where t_r is the failure time corresponding to the r -th failure. It is also usually assumed that the failed units are not replaced in the experiment. This hybrid censoring scheme, which was originally introduced by Epstein [23], has been used quite extensively in reliability acceptance test (Standard [59]).

Interval Censoring

Sometimes exact times of failures are not known; only an interval of time in which the failure occurred is recorded. This type of censoring scheme is known as interval censoring, and the situation is shown in Fig. 22.8.

This type of censoring scheme is used in survival analysis performed in medical field to study the efficacy of drugs or treatment of specific disease, etc.

Progressive Censoring

None of the censoring schemes discussed above allow the items to be withdrawn from the testing before the final termination time of the testing. Sometimes units are lost or removed from experimentation before failure. The loss may occur unintentionally, or it may have been designed so in the study. Unintentional loss may be due to accidental breakage of an experimental unit or the individual under study drops out from the study. Sometimes removal is pre-planned or intentional. For example, to free up testing facilities for other experimentation or to save time and cost, the removal of experimental unit may be pre-planned. Intermediate removal may also be desirable when removed items can be used for some other tests. In view of this, a censoring scheme is developed, known as progressive censoring, where tested

items could be withdrawn at different time points before the end of the experiment [18]. In progressive censoring scheme, items under test could be withdrawn at different time points before the end of the experiment. This type of censoring scheme is widely used in reliability and survival analysis.

To incorporate these situations, censoring is done in several stages. Two important progressive censoring schemes are type-I and type-II progressive censoring. In progressive type-I censoring schemes, m censoring times t_1, t_2, \dots, t_m are prefixed, and n units are put on life test simultaneously [8]. At time point t_i remove R_i units, for $i = 1, \dots, m$ of the surviving units randomly from the test. The experiments terminate at t_m with R_m units still surviving. In progressive type-II censoring scheme, n units are put on life test. At the first failure, R_1 units are randomly removed from the remaining surviving units. At the second failure, R_2 units are randomly removed from the remaining $(n - 2 - R_1)$ units. The test continues until the m -th failure, when all remaining R_m units are removed. More examples are provided in [4-7, 18, 53, 62].

The likelihood function for progressive type – 1 interval censoring can be written as (Aggarwala [2]):

$$\begin{aligned} L(\theta) &= [F(t_1)]^{d_1} \times [1 - F(t_1)]^{R_1} \times [F(t_2) - F(t_1)]^{d_2} \\ &\quad [1 - F(t_2)]^{R_2} \times \dots \\ &\quad \times [F(t_m) - F(t_{m-1})]^{d_m} [1 - F(t_m)]^{R_m} \\ &= \prod_{i=1}^m [F(t_i) - F(t_{i-1})]^{d_i} [1 - F(t_i)]^{R_i} \end{aligned} \tag{22.43}$$

where $F(0) = 0$. For any assumed distribution for t , the parameters can be estimated by using this likelihood function.

As suggested in Aggarwala [2], if one parameter exponential distribution is assumed, then:

$$F(t) = 1 - e^{-\frac{t}{\theta}}, \quad t > 0, \theta > 0. \tag{22.44}$$

The corresponding likelihood function can be written as:

$$L(\theta) = \prod_{i=1}^m [e^{-\frac{t_{i-1}}{\theta}} - e^{-\frac{t_i}{\theta}}]^{d_i} \times [e^{-\frac{t_i}{\theta}}]^{R_i} \tag{22.45}$$

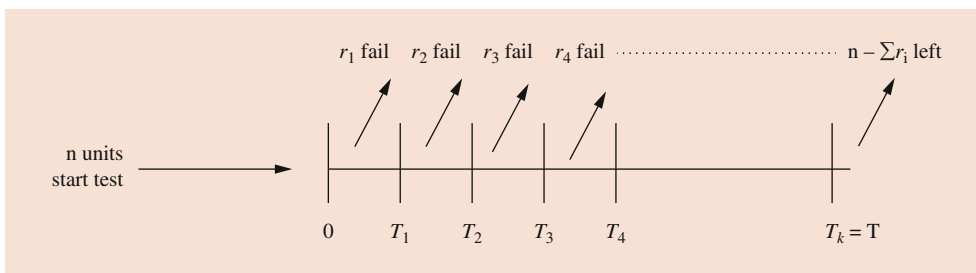


Fig. 22.8 Schematic diagram of interval censoring

If the life time is assumed to have lognormal distribution, as discussed in Roy et al. [55], with PDF

$$f(t) = \frac{1}{t\sigma\sqrt{\pi}} \exp\left\{-\frac{1}{2\sigma^2}(\ln t - \mu)^2\right\},$$

$$t > 0, -\infty < \mu < \infty, \sigma > 0 \quad (22.46)$$

The corresponding likelihood function can be written as:

$$L(\theta) = \prod_{i=1}^m \left[\Phi\left(\frac{\ln t_i - \mu}{\sigma} - \frac{\ln t_{i-1} - \mu}{\sigma}\right) \right]^{d_i}$$

$$\left[1 - \Phi\left(\frac{\ln t_i - \mu}{\sigma}\right) \right]^{R_i} \quad (22.47)$$

$$L(\theta) = \{P[(T \leq C) \cap (t_{i-1} < T \leq t_i)]\}^{d_i} \times \{P[(C \leq T) \cap (t_{i-1} < C \leq t_i)]\}^{r_i}$$

$$= \left\{ \int_{t_{i-1}}^{t_i} f_T(t)[1 - F_C(t)]dt \right\}^{d_i} \times \left\{ \int_{t_{i-1}}^{t_i} f_C(t)[1 - F_T(t)]dt \right\}^{r_i} \quad (22.48)$$

The MLE can be obtained for the parameters of any assumed distribution based on $L(\theta)$.

22.1.5 Accelerated Life Testing

It is important for a manufacturer to study the reliability of components and products, even if it has a very large life span. In such situations, under normal operating conditions, it may take a long time for the components and products to fail and hence calculating reliability based on time to failure data is extremely difficult. Under these circumstances, concept of accelerated life testing is widely used in industry to study the reliability of products. In accelerated life testing, the components and products are subjected to a stress level higher than the usual operating stress so that the failure occurs more quickly. There can be various categories of stress accelerating variables like mechanical, electrical, and environmental (Nelson [48, 49]).

Accelerated Life Testing Models

Under the accelerated life testing scheme, the components and products are subjected to different stress levels, say, s_1, s_2, \dots, s_n . The time to failure of the component or product at each stress levels are then studied. This time to failure data is used to obtain a suitable time to failure probability distribution. Further, the parameters of the selected distribution are estimated. The following can be the basic functional

where $\Phi\left(\frac{\ln t_i - \mu}{\sigma}\right)$ is the CDF for lognormal distribution.

By maximization of $L(\theta)$, one can obtain the MLE of the respective parameters. Similarly, for any distribution of t , $L(\theta)$ can be defined and MLE of the parameters can be obtained.

Random Censoring in the Intervals

In the censoring schemes discussed so far, it was assumed that the censoring occurs at the end of the inspection intervals. If C is a random censoring time, an observation is censored in the interval $(t_{i-1}, t_i]$ if $t_{i-1} < C \leq t_i$, and $C \leq T$ (Meeker and Escobar [40]). Similarly, an observation is a failure in that interval if $t_{i-1} < T \leq t_i$ and $T \leq C$. Then the likelihood function can be written as:

relationship between the normal operating conditions and stress conditions applied to the component or product:

The time to failure, $t_0 = A_F t_s$, where t_0 is the time to failure of the product under usual operating and environmental conditions and t_s is the time to failure under stress conditions and A_F , is defined as the acceleration factor. The cumulative distribution function under normal operating condition is:

$$F_0(t) = F_S\left(\frac{t}{A_F}\right) \quad (22.49)$$

The probability density function is:

$$f_0(t) = \left(\frac{1}{A_F}\right) f_S\left(\frac{t}{A_F}\right) \quad (22.50)$$

The hazard function is:

$$h_0(t) = \left(\frac{1}{A_F}\right) h_S\left(\frac{t}{A_F}\right) \quad (22.51)$$

The most commonly used parametric models are exponential and weibull models.

Exponential Model

In this case, we assume that the time to failure under stress conditions is exponentially distributed with constant failure rate λ_S . Then the cumulative distribution function (CDF) at stress S is:

$$F_S(t) = 1 - e^{-\lambda_S t} \quad (22.52)$$

The CDF under normal condition is:

$$F_0(t) = F_S\left(\frac{t}{A_F}\right) = 1 - e^{-\frac{\lambda_S t}{A_F}} \quad (22.53)$$

The failure rates are related as:

$$\lambda_0 = \frac{\lambda_S}{A_F} \quad (22.54)$$

Weibull Model

The cumulative distribution function (CDF) at stress S for a Weibull model is:

$$F_S(t) = 1 - e^{-\left(\frac{t}{\theta_S}\right)^{\gamma_S}}, \quad t \geq 0, \quad \gamma_S \geq 1, \quad \theta_S > 0 \quad (22.55)$$

where γ_S is the shape parameter of the Weibull distribution under stress condition and θ_S is the scale parameter under stress condition (Pham [52]). The CDF under normal operating condition is:

$$F_0(t) = F_S\left(\frac{t}{A_F}\right) = 1 - e^{-\left(\frac{t}{A_F \theta_S}\right)^{\gamma_S}} \quad (22.56)$$

The MTTF under normal operating condition is:

$$MTTF_0 = \theta_0^{1/\gamma} \Gamma\left(1 + \frac{1}{\gamma}\right) \quad (22.57)$$

and the failure rate under normal operating conditions is:

$$h_0(t) = \frac{\gamma}{A_F \theta_S} \left(\frac{t}{A_F \theta_S}\right)^{\gamma-1} = \frac{h_S(t)}{A_F^\gamma} \quad (22.58)$$

22.1.6 Repairable System Reliability

Consider a repairable system that is put into operation at time $t = 0$. When the first failure happens due to failure of any subsystem or component, it will be brought back to a functioning state after suitable repair or replacement of the component or subsystem. It is assumed that the repair time is negligible in this case. Further, when the second failure occurs, again the system will be repaired and brought back to service. This process continues for all the failures which have occurred in the system. Thus, we get the data on a sequence of time to failure. Let t_1 be the time to first failure. When this failure happens, the system will be restored to a functioning state. The repair time is assumed to be negligible. The second failure will occur at time t_2 and so on. Thus, we get a sequence of failure times t_1, t_2, \dots . Let us define T_i as the time between $(i - 1)$ -th and i -th failure, for $i = 1, 2, \dots$, where t_0 is

considered as 0. The variables T_1, T_2, T_3, \dots , are known as inter-arrival times (time between failures).

This sequence of inter-arrival times T_1, T_2, \dots , will generally not be independently and identically distributed (i.i.d) unless the system is brought back to a state of as good as new condition. Also, for i.i.d assumption, it is necessary that the operational and environmental conditions remain constant throughout the whole period of operation.

Consider a multicomponent repairable system in operation. Depending on how many of its components are in operation, it will have a number of possible states. The state of the system at time t is denoted by a variable $X(t)$. This stochastic process $\{X(t), t \geq 0\}$, is defined as a Markov process and is characterized by its lack of memory property. This means that, if a Markov process is in a state j at time t , we will get no further information about its future states by knowing the past performance of the process up to time period t . Let $N(t)$ be the number of failures in the interval $(0, t]$. Then, the stochastic process $\{N(t), t \geq 0\}$, is called a counting process.

A stochastic process $\{N(t), t \geq 0\}$ is defined as a counting process if $N(t)$ satisfies:

- $N(t) \geq 0$
- $N(t)$ is integer valued
- If $s < t$, then $N(s) < N(t)$
- For $s < t$, $[N(t) - N(s)]$ represents the number of failures that have occurred in $(s, t]$.

Depending on the behavior of $N(t)$, one can determine whether the system is deteriorating or improving. If $N(t)$ is plotted against time t , and if the graph tends to be convex, it indicates that the system is deteriorating. The graph will show concave pattern when the system performance is improving. If $N(t)$ is approximately linear, the system is steady.

Example 7 The failures times (in continuous operating hours) is as follows: 177, 242, 293, 336, 368, 395, 410.

Example 8 The failures times (in continuous operating days) is as follows: 1.0, 4.0, 92.0, 252.0, 277.0, 277.5, 284.5, 374, 440, 444, 475, 536, 568,744, 884, 904, 1017.5, 1288, 1337, 1338, 1351, 1393, 1412.

Some Properties of $N(t)$

Independent Increments

A counting process $\{N(t), t \geq 0\}$ is said to have independent increments if for $0 < t_1 < \dots < t_k, k = 2, 3, \dots$, $[N(t_1) - N(0)], [N(t_2) - N(t_1)], \dots, [N(t_k) - N(t_{k-1})]$ are all independent random variables. In that case, the number of failures in an interval is not influenced by the number of failures in any strictly earlier intervals.

Stationary Increments

A counting process $\{N(t), t \geq 0\}$ is said to have stationary increments if for any two disjoint time points $t > s \geq 0$ and for any constant $c > 0$, the random variable $[N(t) - N(s)]$ and $[N(t + c) - N(s + c)]$ are identically distributed. This means that the distribution of the numbers of failures in a time interval depends only on the length of the interval and not on the intervals distance from the origin.

Availability/Maintainability

Availability and maintainability are two important aspects of any reliability study.

Availability

Availability is the ability of the system to perform required functions over the life of the system. It is the probability that a system is not failed or undergoing a repair action when it needs to be used. It is the proportion of the system “up-time” to the total time (up + down) over a long period.

$$\text{Availability} = \frac{\text{Up time}}{\text{Up time} + \text{down time}} \quad (22.59)$$

Maintainability

Maintainability refers to the measures taken during the development, design, and installation of a manufactured product that reduce required maintenance, tools, logistic cost, and skill levels and ensures that the product meets the requirements for its intended use.

Maintainability is the ability of a component to be retained in, or restored to, a specified condition when maintenance is performed using prescribed procedures and resources. Maintainability is a characteristic of design and is essentially a measure of the ease with which the system can be maintained, because maintainability is often expressed as mean time to repair (MTTR), or how quickly the system can be restored to mission function. Hence, maintainability can be defined as the probability that a device or a system that has failed will be restored to operational effectiveness within a given time [22]. MTTR is defined as:

$$MTTR = \frac{\sum_{i=1}^m \lambda_i T_i}{\sum_{i=1}^m \lambda_i} \quad (22.60)$$

where m is the total number of units and T_i is the corrective maintenance or repair time needed to repair unit for $i = 1, 2, \dots, m$. λ_i is the constant failure rate of unit $i = 1, 2, \dots, m$. Maintainability functions are used to predict the probability that a repair, beginning at time $t = 0$, will be accomplished in a time t . For any identified distribution for repair time, the maintainability function can be estimated.

The maintainability function, $m(t)$, for any distribution is expressed by:

$$m(t) = \int_0^t f_r(t) dt \quad (22.61)$$

where t is time and $f_r(t)$ are the probability density function of the repair time. If the repair time, t follows an exponential distribution with mean time to repair, MTTR, then:

$$\begin{aligned} m(t) &= \int_0^t \left(\frac{1}{MTTR} \right) \exp \left(-\frac{1}{MTTR} \right) dt \\ &= 1 - \exp \left(-\frac{1}{MTTR} \right) \end{aligned} \quad (22.62)$$

Similarly, if the repair time, t , follows lognormal distribution with mean of natural logarithm of t as β and standard deviation of natural logarithm of t as σ , with pdf:

$$f_r(t) = \frac{1}{t\sigma\sqrt{2\pi}} \exp \left\{ -\frac{1}{2\sigma^2} [\ln t - \beta]^2 \right\} \quad (22.63)$$

then

$$m(t) = \frac{1}{\sigma\sqrt{2\pi}} \int_0^\infty \exp \left[-\frac{1}{2} \left(\frac{\ln t - \beta}{\sigma} \right)^2 \right] dt \quad (22.64)$$

22.1.7 Verification and Validation

Verification and validation activities, in short, V&V, are applicable for hardware and software products and also for processes. However, for hardware, the V&V activities are conducted mainly to check whether the design for the product has been correctly carried out. For important medical devices where both hardware and software form part of the device, V&V are to be done both for hardware and software. V&V activities for most software, whether critical or not, are to be done. Hence, verification and testing of software is discussed separately in a later section.

Verification and Validation for Hardware

Verification and validation for hardware is applicable to products which are newly developed and brought to the market. Verification activities mainly ensures that the design for the device is right, whereas, the validation is meant for checking and ensuring that the design is made for the right device that customers will be satisfied with. FDA (<https://www.accessdata.fda.gov/scripts/cfdocs/cfCFR/CFRSearch.cfm?fr=820.3>) defines verification as confirmation by examination and provision of objective evidence that specified requirements have been fulfilled and

validation as confirmation by examination and provision of objective evidence that the particular requirements for a specified intended use can be consistently fulfilled.

Design verification may be carried out using various methods among which inspection, analysis, and tests are more popular. One may refer to FDA guidelines for other methods as well (<https://www.fda.gov/media/116573/download>, Design control guidance for medical device manufacturers).

V&V of design has a lot to do with design inputs. Customer requirements, applicable legal and regulatory requirements, relevant earlier design output which may be used in the current design, feedback from marketing and other departments about an earlier similar design and appropriate guidelines/ procedures or standards etc. form the input to the design process. Inputs for design can also come from the discussion with peer groups. The output of a design and development process could be in the form of a drawing, specification, or results of the analysis. Design output could also be a calculation itself. The ISO 9001 quality management system standard has given guidelines for sufficient control of all such important activities right from its first version released in 1987 [14]. It is also a requirement of the standard that the design or development process output must be documented and be expressed in terms that can be verified as conforming to the input requirements. The users of design output like the production or purchase processes can also sometimes provide important inputs for the design process.

The ISO 9001 standard has categorically stressed upon a planned design review process with appropriate traceability issues to be addressed in all its versions. Two of the important activities under design review is design verification, and validation for all new products that are developed.

Through design and development verification process one needs to ensure that the output of the design and development process meets the requirements given as input to the design process. It is also expected that verification should be planned process to be undertaken by competent personnel. However, there may be several ways for such verification to be conducted by an organization. Verification activities are required for all purchased products as per ISO 9001 standard. Validation of a process is also a must as per the standard, if the product coming out of the process cannot be checked immediately [14].

Validation of design can often be performed by customers or customers representatives. One needs to keep in mind that a proper design evaluation can only be done under user operating conditions. For example, for a motor vehicle, the design would be validated by building a prototype model and driving it over typical road conditions that a variety of customers are expected to use.

Any changes in the design that may be required because of several reasons, calls for a V&V to be conducted again.

22.2 Software Reliability

22.2.1 Introduction

The subject called software reliability was developed as a separate topic of research only in the early seventies of the last century. Most of the people used to consider software as a service giving device and hence were not interested in knowing how its reliability can be measured. It was only after a lot of problems faced with software which started slowly replacing the hardware in developing new devices that researchers and software specialist understood the importance of studying software reliability. But it was known that the theory, that are developed for hardware, cannot be straightaway used for software. Some of the differences between hardware and software are listed in Fig. 22.9. It is, however, noted that day by day, the importance of software will increase and the definition of software reliability can be borrowed from hardware reliability, though the treatment for finding reliability of a software needs to be different.

Under such a scenario, Jelinski and Moranda [29] came out with a very simple model to find out reliability of a software. This model is discussed thoroughly in a later section since their paper throws open the path of thinking about how software reliability can be measured. Subsequent to this development several questions were raised about the sources of uncertainty that is coming into the software, when we know for certain that software does not deteriorate with time unlike hardware items. In fact the term software reliability was not universally accepted, since it is argued that reliability implies that there is some uncertainty about the outcomes, whereas a program either contains one or more error in which case the probability of failure in certain circumstances is unity or it contains no error, in which case the probability of failures is zero.

Considering users point of view, a software output is uncertain since the user will observe software failures at some time point and he would not know when the next failure will occur. Hence, for the software user uncertainty remains.

A software basically transforms an input in the form of data to one or more outputs. Let the input space (totality of all possible inputs) be denoted by I_{total} and that of the output space be O_{total} . The operational profile of the user will however determine the probabilities of selection of different inputs during execution. It is quite likely that different users of a program have different operational profiles.

A software failure occurs when with an input or sometimes it may be called as a case or a test case (during testing of the software); the software is run and the output obtained does not match with the specified output for the case. Let I_F denotes the set of such cases for which the system failures occur. Detection of failures, however, is a nontrivial task [36]. Often

Software reliability	Hardware reliability
1. Without considering program evolution, failure rate is statistically non-increasing.	1. Failure rate has a bathtub curve. The burn-in state is similar to the software debugging state.
2. Failure never occurs if the software is not used.	2. Material deterioration can cause failures even though the system is not used.
3. Most models are analytically derived from assumptions. Emphasis is on developing the model, the interpretation of the model assumptions, and the physical meaning of the parameters.	3. Some distributions are fitted to failure data. The selection of the underlying distribution is based on the analysis of failure data and experiences. Emphasis is placed on analyzing failure data.
4. Incorrect logic, incorrect statements or incorrect input data cause failures. This is similar to design errors of a complex hardware system.	4. Material deterioration, random failures, design errors, misuse and environment are the causes of failures.
5. Increasing the testing effort and correcting detected faults can improve software reliability. Reliability tends to change continuously during testing due to the addition of problems in new code or to the removal of problems by debugging errors.	5. Better design, better material, applying redundancy and accelerated life testing can improve hardware reliability.
6. Software repairs establish a new piece of software.	6. Hardware repairs restore the original conditional.
7. Software failures are rarely preceded by warnings.	7. Hardware failures are usually preceded by warnings.
8. Software components have rarely been standardized.	8. Hardware components can be standardized.
9. Software essentially requires infinite testing.	9. Hardware can usually be tested exhaustively.

Fig. 22.9 Difference between software reliability and hardware reliability

researchers assume that each software failures is caused by only one fault. With the same input space I_{total} , if we randomly choose a case and run it through two programs or software which are made based on the same specification but two different sets of developers; in one the output may match with the expected output whereas for: the other it may result in a software failure. So, apart from the uncertainty about the next input that the software is going to face, probabilistic concepts for software reliability study is valid since the group of developers of the software also contributes to the quality of the software. In a similar way, efficacy of testing and debugging personnel could contribute to the uncertainty in software quality. Broadly software quality can be characterized by two different groups; one for operation of the software and the other for product transition. The following figure (Fig. 22.10) gives the taxonomy of software qualities. However, it is quite obvious that unless the software is reliable, or in other

words, unless it does what we wanted it to do, no matter how good it is with respect to other quality characteristics, the software reliability is the most important quality characteristics.

In satellite launching process, it takes several phases of what is known as rocket technology to lift the satellite to the required height in space, but finally the placement of satellite in the right orbit is done through software. Assume that all the rockets in different phases work as per the requirement, but the software finally fails to deliver its job, then the whole launching process becomes a failure. This also shows the importance of reliability of the software. In fact, in all such modern, sophisticated, and complicated processes software plays a major role; many a times a software replaces a hardware device also. Hence reliability of software has become an interesting topic of research since the 1970s of the last century. There are examples galore, which shows that a small

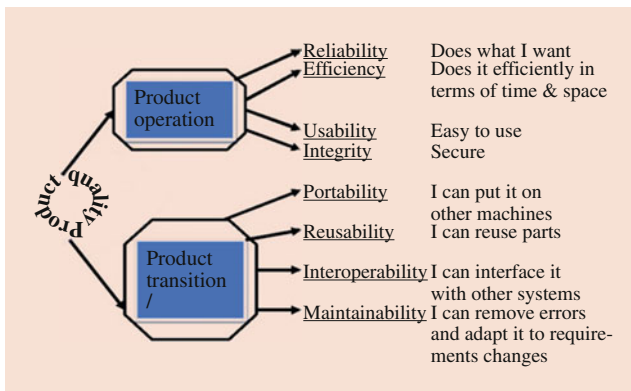


Fig. 22.10 Taxonomy of software qualities

fault in a software had caused havoc in the process. Several such examples can be found in Pham [52] and in many other books. This also shows the importance of software quality characteristics, particularly software reliability. Normally, software is validated using one of the two methods:

- Proving the correctness of the software and
- Testing the software

Proving the correctness of the software is a formal and mathematical process. Methods of proving the correctness of a software include inductive assertion, symbolic execution, and the method of functional verification [34]. In all these methods, a sequence of logically related statements is to be proved. This final statement is usually the output specification statement [26].

Software Testing

Software errors or bugs can arise from the specification, the software system design and from the coding process. Ideally a software should be built in such a way that no bugs are produced during the above three stages of software production [42]. Despite all our efforts to make a reliable software, it is a normal practice to check and test the software before it is released for usage. Formal program checking involving the design team and independent people is called a structured walk through or a code review. An essential part of the software development process is to test the software in such a way that it ensures that the software will operate satisfactorily over the range of possible input conditions. Generally software testing is considered as a planned and disciplined activity so that existence of errors or bugs in the software is minimized. Since a software is a human intensive product, any reasonably large software may carry bugs, even after intensive testing, since testing alone may not be able to guarantee that no bugs will be present in the marketed software. A software normally has several modules and a

full software run may need several module interfacing, which remains a difficult area for software testing.

Software testing group faces several limitations. On the one hand, it is not practical to test a reasonably complex program exhaustively, because the cost and effort for the same could be exorbitant. Software can effectively be considered as having several program paths and testing all the possible paths for the existence of bugs, could be ideal, but remains impractical. Hence, a test strategy needs to be developed wherein the test cases are selected in such a way that will ensure successful operation of the software under the likely range of input conditions without having a cost overrun.

To summarize, a failure is defined as a departure of the output through software operation from requirements. Obviously, a failure occurs when there is some error or fault in the software. This fault is known in the software field as a bug. The process of detecting and removing bugs is known as debugging. For our purposes we assume that when testing of software starts all obvious bugs are removed. This helps us to concentrate and use our resources for detecting and removing nontrivial bugs. We now discuss a bit more detail about software testing since testing is an important activity for evaluating reliability of software. We will use bug, fault, and error in software interchangeably referring to the one and the same thing.

Any mistake in a stage of higher order is going to be more costly than a mistake in a stage of lower order. From the customer point of view, a good software product will have all the eight quality characteristics discussed earlier. All of these requirements should be placed in the very first stage of software development and the stage of understanding customer requirements. The specification or design stage should follow only after the first stage is complete. A good programming technology desires that the specification should not only be developed for the software product but also, at the same time, the specification for testing and validating the software product. Cho [17] calls this as concurrent software design and test design. He also suggests concurrent software implementation and test implementation.

The present-day devices are mostly built with some software as part of the system. Due to the specific nature of software failures, which is not affected by fatigue and which does not fail if it is kept unused for sometime unlike hardware, replacing hardware by software is also very popular. It may be noted that each copy of a software is exactly same as the original. This also helps in avoiding the inherent variation that hardware copies have. Storage of software also become relatively easier and mostly cost effective compared to hardware. Hence, performing functions with software may lead to less complex but more robust systems in the modern-day devices.

Measurement of software reliability is important for the following reasons also.

- Software reliability measures and the information gathered through their use may be exploited to enhance the reliability of the software before releasing it in the market.
- Such measures may be used to predict future behaviors of the software.
- Such measures may be used to monitor the progress of software testing or the operational performance of software using continually updated estimates of current reliability.

The reliability of a unit for a given time is expressed as the probability that no failure occurs in the interval $[0, t]$. Failure for a software occurs when the software results in an output which deviates from the expected output given a particular input. Such failure occurs due to an error or fault in the software. Software reliability at time $t = 0$ is defined to be 1. A couple of problems related to software reliability are

- Determining optimal time for software testing.
- Choosing a suitable model for estimating reliability of software.
 - A Software Developer Dilemma

To understand software reliability better, we have to understand the very process of generating a software. Broadly, like any other product, software when considered as a product, goes through a few stages like:

- Understanding customer requirements
- Developing specification
- Implementation or making the software as per specification
- Validation and
- Delivery which includes packaging and handling also.

Essentially there are three different ways to pursue highly reliable software. They are:

- To avoid the occurrence of faults in the design and development of the program.
- To make use of fault tolerant structures.
- To remove faults during testing phase.

Since software testing is an important activity in the life cycle of a software, a challenging problem being faced by the software manufacturing companies is when to stop testing of the software? There are several authors who worked on this problem (e.g., Chakraborty et al. [15]). We will discuss more about this area in a later section. Also, since several hundreds of software reliability models (SRMs) have already been developed, a common dilemma with the software companies is which model to be used for their particular case.

22.2.2 Software Reliability Models

The very first published work on software reliability is attributed to Jelinski and Moranda [29], in short JM model. The fact that software reliability needs different treatment is quite clear from this model itself. Let us assume that the data available are in terms of time between failures, t_1, t_2, \dots, t_n (assuming that there are n observations taken on a continuous time scale and one has observed the time points $T_i, i = 1, 2, \dots, n$ when the i -th bug is detected). Note that $t_n = T_n - T_{n-1}$. Assume that each of the N faults in the software system will cause a failure after a time which follows exponential distribution with parameter λ , λ being the failure rate of the software. It is also assumed that software faults are independent of each other and are equally likely to cause a failure. It is further assumed that the failure rate λ , at any time point is proportional to the fault content of the software. The authors then assumed that a detected fault is corrected with certainty in a negligible amount of time and no new faults are introduced in the process. Under this set of assumptions, it is clear that the log likelihood function for the observed data t_1, t_2, \dots, t_n is given by

$$\ln L = \sum_{i=1}^n \ln \{\phi(N - i + 1)\} - \sum_{i=1}^n \{\phi(N - i + 1)t_i\} \quad (22.65)$$

where ϕ is the proportionality constant. JM model considers N and ϕ as unknown parameters to be estimated from data. Assuming that N and ϕ are continuous variables, the log-likelihood function is used to find out the partial derivatives with respect to N and ϕ separately. Setting the partial derivatives equal to zero, one gets:

$$\phi = \frac{n}{\sum_{i=1}^n (N - i + 1)t_i} \quad (22.66)$$

and

$$\sum_{i=1}^n \frac{1}{N - i + 1} - \frac{n}{N - c} = 0 \quad (22.67)$$

where,

$$c = \frac{\sum_{i=1}^n (i - 1)t_i}{\sum_{i=1}^n t_i}$$

Though the first published work on software reliability by Jelinski and Moranda was simple to understand, there were a few fundamental mistakes in the model which had been unnoticed for a long time. The unknown constant N in the model being the number of bugs in the software is a discrete

number. Hence taking the partial derivatives with respect to N is not actually valid. However, this model was path breaking which helped many other models to follow. One may refer to Musa et al. [45] to get details of various models developed in the initial years, followed by Chakraborty and Arthanari [13]; Pham [52]; Yang and Xie [63] and many others to know about the software reliability models that were developed later.

The software reliability problems can be broadly classified into two:

1. When one should stop testing software?
2. What is the reliability of the software at any given point in time?

The first problem is also known as optimum testing time problem, whereas the second problem is known as reliability growth problem. For each such problems, several different models are available. The optimum software testing time problem can be based on several different criteria including

1. A threshold value of the reliability of the software to be achieved; typically such software are known as critical or very critical software;
2. The optimal stopping time is based on some cost or reward considerations.

Chaktaborty and his team [15, 16] among many others give the developments in the area of software reliability models to find out optimum software release time. Some authors like Nayak [47] questioned the conventional time-between failure data. Following Nayak [47], Chakraborty and Arthanari [13] showed that the data that are collected particularly for critical and large software are discrete in nature. This shift in data type helped developing several models which assumed a discrete set up. In such a case an input when is used for a software result in either a “success” meaning a bug is detected; or a “failure” meaning the output is as expected.” These terminologies are appropriate since the objective of software testing is to find out as many bugs as possible, so that after debugging the identified bugs, the reliability of the software can be enhanced. Also, in most of the software reliability models, it is assumed that the time-between-failure distribution is exponential. Chakraborty studied this phenomenon and finally concluded that the exponentiality assumption for failure time distribution of a software is a valid assumption.

22.2.3 Software Reliability Growth Models

Time Between Failure Models

In software reliability assessment, this class is one of the oldest. It is expected that during modelling of time between the

failures, the successive time between failures will gradually increase as faults are being removed from the system, and no new faults are introduced during correction. The above statement may not be exactly true, for a given set of observed values since failure times are random and observed values may fluctuate.

Let the time between $(i - 1)$ -th and i -th failures be T_i , a random variable and independent of each other. Usually, it is assumed that T_i follows a distribution with parameters, which depends on the remaining faults after $(i - 1)$ -th failure. Since the faults are detected and debugged from the system, it is assumed that the distribution considered reflects the improvement in the quality of the software. J.M model is one of its kind, which is already explained above. Given below are some of these kinds of models (Fig. 22.11).

Schick and Wolverton (SW) Model

This model (Schick and Wolverton [56]) is based on the same JM model assumptions except that the hazard function is assumed proportional to the time elapsed since the last failure as well as to the current fault content of the program and is given by:

$$Z(t_i) = \phi(N - (i - 1))t_i \quad (22.68)$$

where the various quantities are as defined above. Note that within each failure interval, the above hazard rate is linear with time. It returns to zero when a failure is occurred and again increases linearly but at a reduced slope as the decrease in slope being proportional to ϕ . In 1978 a modification of the above model was proposed, and in test time the hazard function is assumed to be a parabolic and is given by:

$$Z(t_i) = \phi[N - (i - 1)](-at_i^2 + bt_i + c) \quad (22.69)$$

where a , b , and c are constants and the other quantities are same as defined above. The above hazard function consists of two components. The first is the hazard function of the JM model, and the superimposition of the second term indicates that as the test time accumulates within a testing interval, the likelihood of a failure occurring increases rapidly. At failure times $t_i = 0$, the hazard function is proportional to that of the JM model.

Goel and Okumoto Imperfect Debugging Model

Most of the software reliability models, including the ones that are discussed so far, assume that the fault that is detected through testing is completely removed during debugging. But, it has been observed that many a times, this assumption is not valid. One way to model such a phenomenon is to take resort to an imperfect debugging model developed by Goel and Okumoto [27], which is an extension of the JM model. Goel and Okumoto [27] considered that the number of faults X_t , in the software at time t is represented by

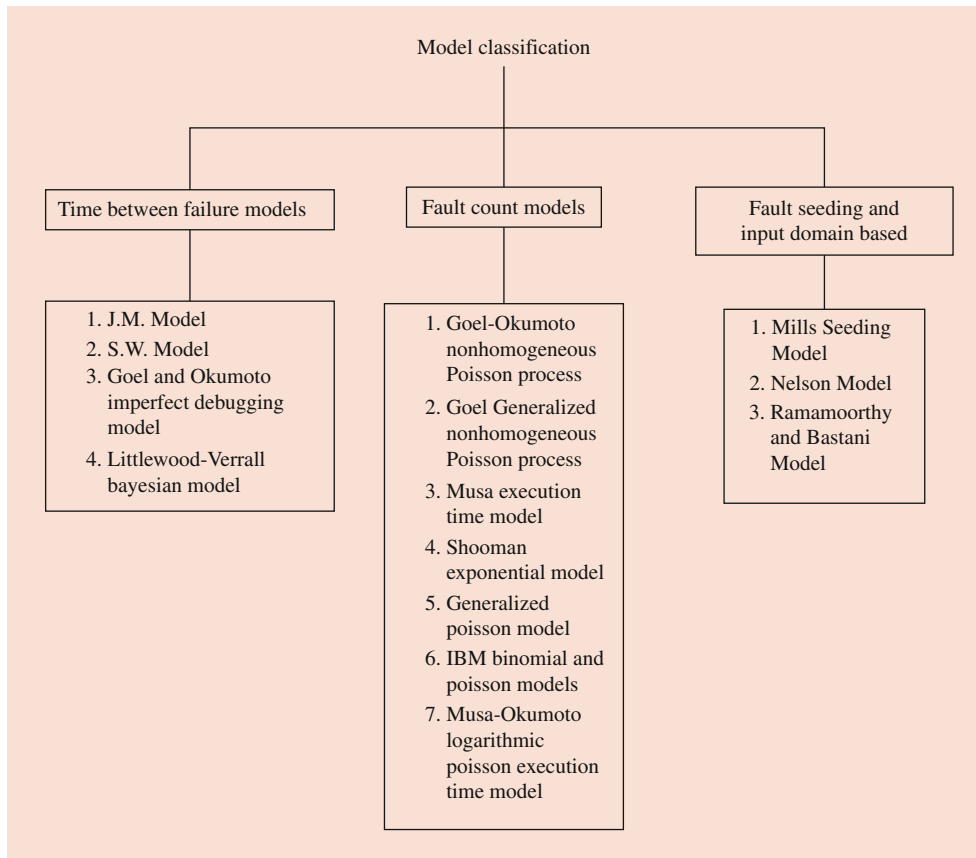


Fig. 22.11 Classification of Growth models

a Markov process, wherein the transition probabilities are based on the probability of perfect debugging. The time intervals between the transitions of X_t s are assumed to be following an exponential distribution whose failure rates are considered as a function of the number of faults that exists in the software at time t . The hazard function during the interval between the $(i - 1)$ -th and the i -th failures is given by:

$$Z(t_i) = [N - p(i - 1)]\lambda, \quad (22.70)$$

where, N is the initial fault content of the system, p is the probability of imperfect debugging, and λ is the failure rate per fault.

Littlewood-Verrall Bayesian Model

Littlewood and Verall [37] took a different approach in the development of a model. They argued that software reliability in terms of the number of errors should not be specified in a program. Specifically, in their model, the times between failures are assumed to follow an exponential distribution but the parameter of this distribution is treated as a random variable with gamma distribution, viz.:

$$f(t_i | \lambda_i) = \lambda_i e^{-\lambda_i t_i} \quad \text{and} \quad (22.71)$$

$$f(t_i | \alpha, \psi(i)) = \frac{(\psi(i))^\alpha \lambda_i^{\alpha-1} e^{-\psi(i)\lambda_i}}{\Gamma(\alpha)} \quad (22.72)$$

In the above, $\psi(i)$ describes the difficulty of the programming task and the quality of the programmer. It is claimed that in this model the failure phenomena can be explained in different environments by taking different forms for the parameter $\psi(i)$.

22.2.4 Fault Count (FC) Models

Fault count models deal with the modelling of the number of faults or failures in the testing intervals. It is expected that after the removing of the faults from the given system, the number of faults will decrease. Here the time used can be calendar time, CPU time, or others. In these models, the time intervals are taken as fixed but the number of failures vary for individual intervals. The concept behind most of these models is that the distribution considered here is Poisson distribution with different parameters for different models. Some common assumptions are like, each testing intervals should be independent, and within the interval, testing should be homogeneous. During the detection of the faults, the

number of faults should be independent in nonoverlapping intervals. Some of the models of this class are given below:

Goel-Okumoto Nonhomogeneous Poisson Process Model (Goel and Okumoto [28])

The authors here assumed that the software system fails at random time points because of the presence of faults in the software. They defined $N(t)$ to be the cumulative number of failures observed by time t . They then modeled $N(t)$ as a nonhomogeneous Poisson process (NHPP) where the failure rate is considered as a function of time. They had the opportunity to study several real-life failure data from many systems. This experience helped them to propose the following model.

$$P[N(t) = y] = \frac{m(t)^y}{y!} e^{-m(t)}, y = 0, 1, 2, \dots \quad (22.73)$$

where $m(t)$ = expected number of failures observed by time $t = a(1 - e^{-bt})$ and $\lambda(t)$ = failure rate = $m'(t) = abe^{-bt}$.

It is easy to note that a = the expected number of failures observed eventually and b = the fault detection rate per fault. Unlike the Jelinski-Moranda [29] and many other models, Goel and Okumoto allowed the number of faults to be detected as a random variable. The value of this random variable may depend on several factors, including the environmental factors.

A big advantage of the nonhomogeneous Poisson process model is that by suitably varying the mean value function, $m(t)$, many other models may be developed.

In an earlier attempt, Schneidewind [57] studied the number of bugs detected during a given time interval using a somewhat different approach. In his approach he considered the failure process to be a nonhomogeneous Poisson process which has an exponentially decaying intensity function given by:

$$d(i) = \alpha e^{-\beta i}, \alpha, \beta > 0, i = 1, 2, \dots \quad (22.74)$$

where α and β are the parameters of the model.

Goels Generalized Nonhomogeneous Poisson Process Model (Goel [26])

Further generalization of the Goel-Okumoto NHPP model was proposed by Goel [26], where he considered $m(t) = a(1 - e^{-btc})$, a is the expected number of faults to be eventually detected, and b and c are constants that reflect the quality of testing. The main formulation of the model remains the same as given in (22.73), that is:

$$P[N(t) = y] = \frac{m(t)^y}{y!} e^{-m(t)}, y = 0, 1, 2, \dots \quad (22.75)$$

The failure rate for the model proposed by Goel is given by:

$$\lambda(t) \equiv m'(t) = abe^{-bct} t^{c-1} \quad (22.76)$$

It is generally assumed that reliability of software improves with testing, since as the testing and debugging increases, it is expected that the remaining number of bugs in the software will decrease. Hence, most of the times between failures and failure count models assume software reliability growth, which in a way assumes that a software system exhibits a decreasing failure rate pattern during testing. However, in reality, it may be observed that in many testing situations, the failure rate first increases and then decreases. In order to model this increasing/decreasing failure rate process, Goels proposed generalization of the Goel-Okumoto NHPP model may be more appropriate.

Musa Execution Time Model (Musa [43])

In this model, the focus is on the number of failures in a specified execution (of the software) time interval. The assumptions for the model are similar to that of the Jelinski-Moranda model. The hazard function, $z(\tau)$, is given by:

$$z(\tau) = \phi f(N - n_c) \quad (22.77)$$

where τ is the execution time utilized in executing the program up to the present; f is the linear execution frequency (average instruction execution rate divided by the number of instructions in the program); ϕ is a proportionality constant, which is a fault exposure ratio that relates fault exposure frequency to the linear execution frequency; and n_c is the number of faults corrected during $(0, \tau)$.

Generalized Poisson Model

This is a variation of the NHPP model of Goel and Okumoto and assumes a mean value function of the following form.

$$m(t_i) = \phi(N - M_{i-1})t_i^\alpha \quad (22.78)$$

where M_{i-1} is the total number of faults removed up to the end of the $(i - 1)$ -th debugging interval, ϕ is a constant of proportionality, and α is a constant used to rescale time t_i .

IBM Binomial and Poisson Models

Brooks and Motley [10], probably for the first time, guessed that the software testing process may be considered as a discrete process. Quite obviously, the number of detected faults may then be modeled under a Binomial or a Poisson distribution. They proposed, as is normally the case, that the software is to be developed and tested incrementally. Further, they advised their models to be applicable both for the modules, as well as, for the whole software.

Musa-Okumoto Logarithmic Poisson Execution Time Model

Taking cue from the Goel-Okumoto model, Musa and Okumoto [44] developed another NHPP model where the mean value function considered was different. Let τ denote the time, then:

$$\mu(\tau) = \frac{1}{\theta} \ln(\lambda_0 \theta \tau + 1) \quad (22.79)$$

where λ_0 and θ represent the initial failure intensity and the rate of reduction in the normalized failure intensity per failure, respectively. It may be noted that Musa-Okumoto model resembled Morandas geometric de-eutrophication model and may be considered as a continuous version of Morandas model.

22.2.5 Fault Seeding and Input Domain based Models

In this section, we will discuss about a few time-independent models for assessing software reliability. The two approaches are fault seeding and input domain-based models.

In fault seeding models, in a program, a known number of faults are seeded. After testing the number of native faults along with the exposed seeded faults are counted. Applying maximum likelihood estimation and using combinatorics, the number of native faults and reliability of the software are estimated in a program. In case of input domain-based models, the basic idea is to generate a set of test cases from an input distribution. The input domains are partitioned into equivalence classes in various models in this group because the input distribution is difficult to estimate. The classes are related with the program path. The reliability measure is calculated from the number of failures observed during physical or symbolic execution of the sampled test case.

Mills Seeding Model

One of the techniques used in estimating the number of bugs present in a software quite early in the development of software reliability is called the error seeding model (Mills [41]). The procedure consists of generating several faults that are randomly seeded into the software that will be tested. The program is then tested in order to find as many faults as possible. Subsequently, among the faults detected, the number of original faults and the specially seeded faults are counted. Based on this data, a hypergeometric distribution is used to estimate the total original number of faults that were originally present in the software. This technique, however, had been used earlier in similar situations, for example, for estimating the number of fish in a pond or for estimating

wildlife population in a given area. Sometimes these models are also known as tagging models since the seeded faults are tagged for identification purpose.

A modification of this problem where the probability of finding a fault, of either kind, is considered by Lipow [35]. Then, assuming independence of tests, the probability of finding given numbers of indigenous and seeded faults can be calculated. In a further modification of this model, Basin [9] used a two-stage procedure with the use of two programmers which can be used to estimate the number of indigenous faults in the program.

Nelson Model

The Nelson model is one of those input domain-based models where the reliability of the software is determined based on the results obtained by testing the software with a sample of n inputs which are chosen randomly from the input domain set $E = (E_i : i = 1, \dots, N)$ where each E_i is the set of data values needed to make a run. The operational profile of the user is represented by a probability distribution $\{P_i : i = 1, \dots, N\}$, and the n random samples are chosen according to the probability distribution P_i . Assuming that n_e represents the number of inputs that resulted in execution failures, an unbiased estimate of the reliability of the software is given by the following formula.

$$\hat{R}_1 = \left(1 - \frac{n_e}{n}\right) \quad (22.80)$$

However, one may observe that the test cases used during the verification phase or even afterwards may not represent the expected operational profile of the users. This may be the case more often than not. In such circumstances, Brown and Lipow [11] suggested an alternative formula for the reliability of the software, R which is:

$$\hat{R}_2 = 1 - \sum_{j=1}^N \left(\frac{f_j}{n_j}\right) p(E_j) \quad (22.81)$$

where n_j denotes the number of runs sampled from input subdomain E_j and f_j is the number of failures observed out of n_j runs. The basic difference between \hat{R}_1 and \hat{R}_2 is that the former explicitly incorporates the usage distribution or the test case distribution while the latter implicitly assumes that the accomplished testing is representative of the expected usage distribution. Hence, both models assume prior knowledge of the operational profile in terms of a probability distribution.

Ramamoorthy and Bastani Model [54]

This model is also an input domain-based model where the authors considered the real-time, critical, process control systems for estimating reliability. It is expected that for such systems, the reliability should be very close to one. Hence, it

is also important that we have high confidence in the estimate for reliability of the software. Using this model one can get an estimate of the conditional probability that the software will give correct output for all possible inputs given that it is correct for a specified set of inputs. The interesting idea used in the model is that each output from the software gives some stochastic information about the behavior of the software for other inputs which are close to the test case used.

One of the important results in this model can be written as:

P (program is correct for all points in $[a, a+V]$ | it is correct for test cases having successive distances $x_j, j = 1, \dots, (n-1)$)

$$= e^{-\lambda V} \prod_{j=1}^{n-1} 2(1 + e^{-\lambda x_j})^{-1} \quad (22.82)$$

where λ is a parameter which is dependent on the complexity of the source code.

This model allows any test case selection strategy, which is unusual. As is practiced in many organizations, test cases are chosen that run through error-prone constructs. This also helps in reducing testing effort. However, the model concerning the parameter λ needs to be validated experimentally.

22.3 Optimum Release Time Models

A very important part of the software testing problem is when to stop testing the software? (Dalal and Mallows [19]). The main aim of testing a software is to find out as many bugs as possible before the software is released. Also, once a bug is identified, it is debugged and hence the reliability of the software is improved. Several authors have worked over this optimization problem and looked into it from various angles. One of the important constraints that is used to solve the optimization problem is the cost or time constraint. Normally this type of constraints is relevant for commercial software. But there are other types of software as well, like the software used for space research. These are called critical or mission critical software. For such software, achieving a high reliability is more important than the cost of testing it (Dewanji et al. [21]). Also, over the decades the type of test data available got changed, which led to the development of discrete framework in the software reliability arena (Chakraborty and Arthanari [13]). Long before that Nayak [47] proposed to log software data in a different way. A few authors worked on a discrete framework for optimum software release time.

The data that are normally available for mission critical and large software are of the discrete data type. In such cases, for each test data the outcome is known in the form of either a success, meaning a bug is found or a failure, meaning no bug is found with the present test case. Also, there is a major

violation of the assumption that most software reliability models have is as soon as a bug is found it is debugged. Such an assumption remained valid for the initial stages of software reliability developments. However, as the software becomes larger and larger, the developer and testers became two different groups and the assumption started getting violated. This necessitates development of different models for optimum software release time (Chakraborty et al. [15]).

However, majority of the models developed for optimum time for software release still adopt the assumption of immediate debugging in order to make the models relatively less complex. Dalal and Mallows [19] derived an optimum stopping rule for testing of a software prior to release, based on a trade-off between the cost of testing for a longer period and expected loss due to any bugs that may remain in the software after release. Singpurwalla [58] addressed the same problem by maximizing expected utility functions of two different types. The information available during the testing phase is, like many other models, the time between failures. Dalal and Mallows [20] later developed methods for optimum stopping with exact confidence limits on the remaining number of bugs in a software. Nayak [47] proposed for the first time that software testing data should capture test case-based information in order to have more insight into the reliability aspect of the software. Chakraborty [12] proposed a discrete framework for software testing data where the information obtained from software testing is for every test case. The outcome is either a success which indicates detection of a bug or a failure indicating that no bugs are found using the test case. This kind of information are largely logged for large and critical softwares (Dewanji et al. [21]).

Usage of Bayesian concepts in software reliability field has not been regular. Chakraborty and Arthanari [13], Vasanthi and Arulmozhi [61], and Chakraborty et al. [15, 16] are some references where the Bayesian concepts were used to solve the problems. In a discrete framework, Chakraborty and Arthanari [13] show how to find out optimal time of testing. They, however, used the standard assumption that as soon as a bug is identified, it is fixed. This assumption does not hold good for many large and critical software (Dewanji et al. [21]). After dropping this critical assumption, Chakraborty et al. [15] developed the Bayesian methodology to find out the optimum time for software release.

A new twist in the development of the software reliability models is given by looking at the software as a bunch of several paths and sub-paths. Each test case is expected to go through a particular path in the software. The test case can identify a bug if and only if the bug lies on the path that the test case is supposed to traverse. This concept, though not new, gives rise to a newly developing theory of size-biased concept in software reliability field (Chakraborty et al. [16]). The idea seems to be very interesting and opens up new areas of research in the software reliability arena.

22.4 Software Testing and Validation

Testing is a group of techniques to determine the correctness of the application under the predefined script, with an intention to detect a major portion of its bugs, so that they are discovered and debugged, and, at the same time, verified that the software product under consideration is fit for use. However, it does not demonstrate that a product functions systematically under all conditions but it only ensures that it is not working in some specific formats (Kaner [31]). In software testing and management, verification and validation (V & V) are the process of checking that a software system meets specifications and that it fulfills its intended purpose. It may also be referred to as software quality control. In general, the quality assurance team conducts the verification process. This is followed by testing of the software and its validation that are both executed by the testing team.

22.4.1 Introduction to Software Testing

Software testing is a process, or a series of processes, designed in such a manner so as to ensure that computer code does what it was designed to do and that it does not do anything unintended. Any software released in the market should be predictable and consistent, offering no surprises to users.

In the present-day circumstances, due to the advancement in operating systems and other very useful software, which provide some intrinsic, well tested routines that can be made use of during software development itself. This helps programmers/testers to develop better software rather easily without developing the routines from scratch. Graphical user interfaces (GUIs) can be built from a development language library, and since they are pre-programmed objects that have been debugged and tested previously, the need for testing them as a part of a custom application is much reduced (Jorgensen [30]).

22.4.2 Some Basic Definitions

The International Software Testing Qualification Board (ISTQB) has produced an extensive glossary of testing terms and some of them need to be considered before moving on to understanding what is actually software testing (Martin [39]).

Error

The mistakes committed while coding of a program are termed as errors or more appropriately as bugs. Errors tend to propagate and can also be magnified during design or amplified even more during coding.

Fault

A fault can be considered as a manifestation of an error in the program. A fault in the program may appear due to design deficiency, incorrect coding, etc. Quite often faults could be elusive. For example, a fault may exist due to an error of omission on the part of the programmer or even designer.

Failure

A failure occurs when a faulty code is executed. Thus, failures occur in the executable source code or loaded object code. In the process, the expected outcome from the input cannot be obtained.

Incident

Failures in a program may occur without the user or testers apparent knowledge. However, a symptom may be apparent that is connected with a failure which usually provides alarms to the user indicating occurrence of a failure. These symptoms are called incidents.

Test

When a software is executed with some test cases, we say that the software is being tested. Testing of a software, using a test case, may lead to either a failure or a success, where success means demonstration of correct execution. Thus, testing is normally connected with failures and incidents which in turn are connected with errors and faults.

Test Case

A test case has a set of inputs for which the outputs are known. Usually a test case has some identification which helps all concerned to understand why the test case was run and is associated with the program behavior.

Figure 22.12 represents a life cycle model for testing a software. It is to be noted that although the development phases of a software, errors may creep in resulting in faults that may propagate through the remainder of the development process. Also, more errors may be added during the fault resolution step as well.

It is important to note that developing appropriate test cases is a key for effective testing. In fact, the whole of the testing process consists of making a proper test plan, developing effective test cases, which is followed by actually running the test cases as per the plan and finally evaluating the test results.

22.4.3 What Is Software Testing?

Testing of a software is considered to be a process through which the software and its components are evaluated with the

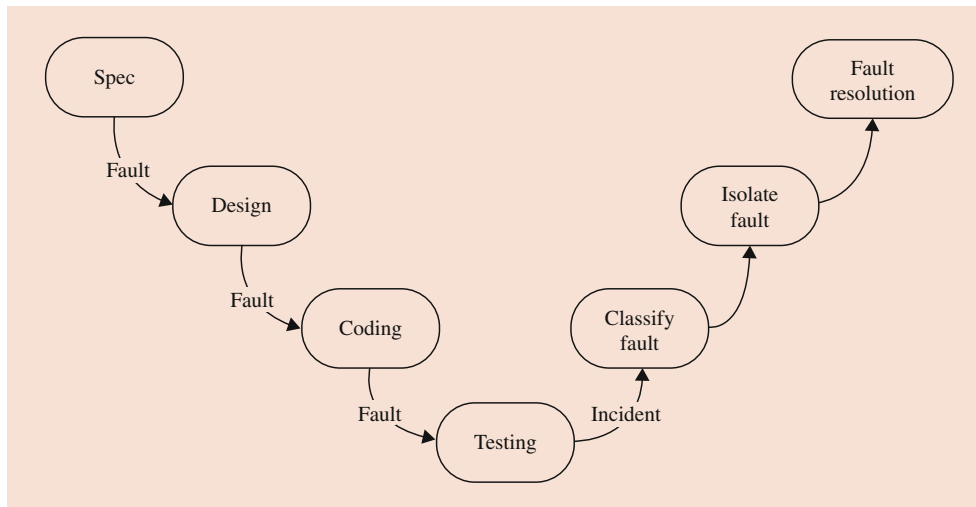


Fig. 22.12 A testing life cycle

aim to find out whether it satisfies the specified and desired requirements or not. In other words, software testing refers to the process of execution of the software based on a plan, in order to identify any gaps, errors or missing requirements in contrary to the actual requirements (Lyu et al. [38]).

According to ANSI/ IEE 1059 Standard, Software Testing can be defined as “A process of analyzing a software item to detect the differences between existing and required conditions (i.e., defects/ errors/ bugs) and to evaluate the features of the software item.

22.4.4 The Psychology of Testing

Commonsense says that if all possible permutation of a software could be tested, it would have been ideal. However, such a possibility is simply remote in most cases. Even if we take a relatively simple program, then also the possible input and output combinations may run into thousands and developing test cases for such a huge number is just impractical. Many software are as such quite complex in nature and trying to test it completely could be economically infeasible.

Also, a software tester must have the proper attitude or vision in achieving the goal towards testing a given software application. One of the primary causes of poor program testing may be the fact that most programmers start with an incorrect definition of the term software testing. They might say that “Software testing is a process of demonstrating that errors are not present.”

But in fact, while software is being tested, the target is to add some value to it. Adding value through testing means enhancing the quality or reliability of the program and hence of the software with finding and debugging of errors. Therefore, the correct approach is to start with the

assumption that the software contains bugs and then test the program to detect as many errors as possible. Thus, a more appropriate definition may be “Software testing is the process of executing a program with the intention of finding errors.”

Human beings tend to be highly goal-oriented, and establishing the proper goal has an important psychological effect. If our goal is to demonstrate that a software is completely non-erroneous, then we will subconsciously be steered towards achieving this goal. On the other hand, if the goal was to demonstrate that a software does possess some errors, then the test data will have a higher probability of finding the same. The latter approach is likely to add more value to the software program as compared to the former (Myers et al. [46]).

22.4.5 Who Does Testing?

Depending on the process and associated stakeholders of the project(s), a person or a group of persons with sufficient experience is chosen to execute the testing process. Testing, as has been said earlier is a planned and a must activity for any software developer. Large IT companies generally engage a set of people who are given the responsibility to test the software against the given requirements. In order to ensure satisfactory output from the software, most organizations start testing at the module and or function level itself, which is known as unit testing. Also, the end user normally checks the software for satisfactory outcomes. Thus, in most organizations the professionals who are associated with the testing process may be designated as software tester, software developer, software quality assurance engineer, quality assurance analyst, project manager/lead, and customer/end user, among others.

22.4.6 The Optimal Time for Testing

Since identifying faults in a software is a nontrivial problem and also it is known that the faults may be created at every stage of the software development process, it is ideal that testing starts right from the first stage called requirement gathering phase and continued till the deployment of the software. This helps in reducing the overall cost associated with the software development life cycle (SDLC). However, the testing strategy may depend upon the development model that is being followed. In many organizations, formal testing is carried out in the testing phase, whereas for others, testing may be conducted at the end of every phase and the whole software is tested at the end (Ammann and Offutt [3]).

In most situations, testing is carried out in different forms at every developmental phase of the software, such as:

- During the requirement gathering phase, the analysis and verification of requirements are considered as testing.
- Reviewing the design in the design phase with the target of improving the design is also considered as testing.
- Testing performed by a developer on completion of the code is also categorized as testing.

It is difficult to determine when one should stop testing, as testing is usually a never-ending process and no one can claim that a software is 100% tested. However, some of these aspects can be considered for stopping the testing process.

- When testing deadlines are met.
- When test case execution is completed.
- When functional and code coverage is completed to a certain extent.
- When the bug rate falls below a certain level with no high-priority bugs being identified further.
- When the management team decides that the given testing can be brought to an end.

Several models for deciding the optimal time for testing a software are discussed in section 22.3.

22.4.7 Black-Box and White-Box Testing

In order to make a software reliable, its testing strategies can broadly be classified as either a black-box testing or a white box testing. In black-box testing strategy, the program is viewed as a black box, whereas for white-box testing, the internal logic and structure of the program becomes the main focus. Black box testing is data-driven or one can say input-output driven whereas white box testing depends on the internal logic (logic-driven) based on which the software has been developed.

As the name suggests, in black box testing, the test data are generated only from the specifications, without trying to know its internal structure. The aim is to find circumstances in which the program behavior does not match with the specifications.

In white box testing on the other hand, the tester examines the internal structure of the software. The test data in this case are generated based on the program logic, even sometimes neglecting the specifications (refer Everett and McLeod Jr. [24]).

22.4.8 To Conclude Testing

Certain principles of testing must be kept in mind before moving on to software validation.

- Testing is the process of executing a program with the aim of finding as many bugs as possible.
- A good test case is considered to be the one that has a high chance of detecting as yet undetected bug(s).
- Hence, a successful test case is the one that detects an as yet undetected bug.

22.4.9 Software Verification and Validation

The Quality System Regulation is harmonized with ISO 8402: 1994, which treats “verification and validation” as separate and distinct terms. However, quite often, many software engineering journals use these two terms interchangeably, or in some cases refer to software “verification, validation and testing (VV&T)” as if it is a single concept with no distinction among the three terms. Boehm succinctly expressed the difference between the two terms as in (refer Pham [51]).

- Verification—Are we building the product right?
- Validation—Are we building the right product?

By verifying the software at each phase of the software development process, we try to provide objective evidence that the design outputs of a particular phase meet all the requirements specified for that phase. It looks for consistency, completeness, and correctness of the software, and its supporting documentation, as it is being developed, and provides support for a subsequent conclusion that the given software under consideration is validated. Software testing is used as one of the several verification techniques intended to assure that software development output meets its input requirements. Other verification methodologies include various static and dynamic analyses code and document inspections, walk through among others (refer Abdeen [1])

Software validation is a part of the design validation for a developed software. It may be formally defined as (refer [60]): *The confirmation by examination and provision of objective evidence that software specifications conform to user needs and intended uses, and that the particular requirements implemented through software can be consistently fulfilled.*

Validation of software, which may be conducted both during as well as at the end of the software development life cycle, is for ensuring that all the expected requirements are satisfied. In many cases, software are used as a part of a larger hardware system. Hence, by validating a software evidence is provided that all software necessities have been correctly and completely catered to and are traceable to the system requirements

A conclusion that software is validated is highly dependent upon comprehensive software testing, inspections, analyses, and other verification tasks performed at each stage of the software development life cycle. Testing of device software functionality in a simulated user environment and user site testing are typically included as components of an overall design validation program for a software automated device.

Verification and validation of software are difficult phases of the software development process, because it is not known to the developer how much evidence can be considered to be enough and also the software cannot be tested forever. Hence, software validation is considered as developing certain level of confidence that the device meets all the requirements and expectations from the users about the software automated functions and features of the device. Different measures which include (i) defects found in specifications, (ii) documented records, (iii) estimates of remaining defects/bugs in the software, (iv) testing coverage, etc. are used to obtain an acceptable level of confidence before a decision is taken to release the software in the market. In terms of some latest developments in the software reliability field, a sufficiently small total remaining bug size [16] may be a better indicator for proving that the software is reliable.

The level of confidence and therefore the level of software validation, verification, and testing effort needed will vary depending upon the safety risk (hazard) posed by the automated functions of the device (refer Freund [25]).

22.4.10 Software Validation as Part of System Design

System functionality is usually implemented in the designing phase of the software. Software requirements, in particular, depend on the overall system requirements as well as the design of the software that has been initially modeled. User needs depend a lot on the expected uses of a finished product,

and these specifications can be addressed by either software, hardware, or a combination of both. Therefore, software validation should be incorporated within the framework of the overall design validation for the system.

In the software requirement analysis phase, user needs and its intended uses based on which the software should be developed are documented. This documented software and system requirements are to be proved to have been satisfied completely during validation of software product. The appropriateness and completeness of both the system requirements and the software requirements should be addressed as part of the design validation process for the device. Thus, software validation includes confirmation of satisfactory conformance to all software specifications and confirmation of traceability to the system specifications of all software requirements. Confirmation is an important part of the overall design validation to ensure that all aspects of the software confirm to user needs and intended uses (refer [60]).

22.4.11 Principles of Software Validation

Some of the general principles that should be considered for the validation of software are defined by (refer Freund [25]).

Requirements

Software requirements analysis phase of the software development life cycle consists of documentation of the requirements from the software and the specifications derived from it forms part of the design phase. These documented versions form the baseline for both verification and validation. Hence, a software validation process cannot be carried out without an established software requirements specification.

Defect Prevention

Quality assurance for software should focus on prevention of introduction of errors during the software development process and definitely not to test quality into the software code that is already written.

Though testing is a part and parcel of any software development process, it cannot ensure detection of all defects in a software code. As such, the complexity of most software prevents it from being exhaustively tested. Software testing alone may not be able to provide sufficient confidence that the software is fit for its intended use, though it is a necessary activity. A better way to develop confidence into the product is to use an ensemble of methods that on one hand will prevent occurrence of errors when the software is being developed and on the other detect as many bugs as possible that has already gone in the software. There is no one best mix of these methods for all software; however, the strategy depends

on several important factors which include the development environment, application, size of project, language, and risk.

Time and Effort

A lot of time and effort is required to build a case that is used to validate a software. In fact, one should prepare for software validation right in the beginning, that is, during design and development planning stage itself. Planned efforts put throughout the software life cycle and the evidence obtained through that should be the basis for finally validating a software.

Software Life Cycle

There are several software life cycle models. All the models aim at developing a software that will ultimately satisfy the intended users. Sufficient documentation and software engineering tasks are required in order to support software verification and validation efforts.

Plans

A software validation plan must necessarily include the ultimate goal of the software validation process. Hence, the plan needs to be defined before the validation process starts. The software validation plan forms an important quality system tool which includes the following: (i) scope, (ii) approach, (iii) resources, and (iv) schedules along with the different types and extent of activities, tasks, and various work items (refer [60]).

Procedures

The software validation process is generally a step-by-step approach which uses some specific procedures to answer the question as to how the software validation effort should be conducted. These procedures also include identification of specific activities or a sequence of activities that need to be followed in order to complete individual validation activities, tasks, and work items (refer [60]).

Software validation After a Change

A seemingly small local change in the software may result in a drastic alteration on the global system, on account of the complexity of software. When any change (even a small change) is made to the software, the validation status of the software needs to be re-established. Any change in the software calls for a validation analysis of not just the particular change that has been effected, but the analysis should determine the extent to which the change has impacted the whole software. Based on this analysis, the software developer needs to carry out a suitable software regression testing in order to ensure that vulnerable portions of the system are not affected adversely. Through design controls and suitable regression testing confidence can be built that the software is validated after a software change (refer [60]).

Validation Coverage

The complexity of the software and issues like safety risk should determine the validation coverage of the software under consideration. Determinants such as firm size or resource constraints cannot be considered to be adequate. The selection of validation activities, tasks, and work items should be adaptable to the complexity of the software design and the risk associated with the use of the software for the specified intended use. For the devices that are not that much vulnerable to risks, only baseline validation activities may suffice. As the risk becomes more and more, additional validation activities need to be fulfilled in order to make up for the additional risk. Validation documentation should be sufficient to demonstrate that all software validation plans and procedures have been completed successfully (refer [60]).

Independence of Review

A number of reviews are essential before final release of the product into the market, but these reviews should not depend on one another. Self-validation is not only extremely difficult, it does not properly serve the purpose. Like most management systems, independent reviews are always better, particularly for complicated and high-risk applications. Some organizations hire from outside a software verification and validation team; however, this may not always work out to be the appropriate one. In most cases, experienced and knowledgeable internal members who are not associated with the design of the software or its implementation are assigned the job of conducting the verification and validation activities. However, for smaller organizations the challenge remains as to how to organize and assign the tasks in order to ensure that the reviews so undertaken are independent.

Flexibility and Responsibility

Like all other quality system management standards, software validation principles also may be implemented in different ways for different software and their applications. The freedom of planning lies with the software developer or the device manufacturer as to how to apply different validation principles in order to demonstrate that the software has been validated.

Software may need to be developed for a wide variety of devices with a wide spectrum of environments having different levels of risk. An example could be the software that are used in medical devices for which FDA regulations are applicable. Such a software could be a component or part or may work as an accessory of a medical device. It could by itself be a medical device, or it may be used in manufacturing, design and development of a system, or it could be used for other parts of a quality management system.

Given an environment, a software may consist of several different components, some of which are developed in-house, some may be purchased off-the-shelf, whereas some may

be contract software or shareware. Also, software components that are developed can be used as application software, operating systems, compilers, debuggers, and configuration management tools, among others.

Validating software in such different environments is a nontrivial task, and hence, it is all the more necessary that software validation process is designed keeping in view all of these software validation principles. The validation process thus developed should normally be equivalent to the safety risk associated with the system, device, or process (refer [60]).

22.4.12 Benefits of Software Validation

Software validation process is an important process step of the overall software development process and it helps build confidence in the product and its automated operations. This process helps in the enhancement of the reliability of the software, which automatically is reflected in decreasing failure rate, less number of recalls and corrective actions required. This also helps in reducing the liability of the software developer and finally helps in minimizing risks to the ultimate users and stakeholders.

It is a known fact that like testing of software, maintenance of software also is an important activity and both the activities take away a large chunk of the cost involved for the entire life cycle of the software. A software validation process which is comprehensive and well documented helps not only to reduce the life cycle cost of the present software; it also helps reducing the long-term cost of development of any subsequent versions of the software (refer Koopman [32]).

In other words, the validation process actually reduces the long-term cost by making it not only easier but less costly, as well, for reliably modifying the software and revalidating software changes.

Though software validation activity may be carried out from different geographical locations involving many different groups, the onus of ultimate responsibility for ensuring that the software is validated, rests on the developer.

22.4.13 Validation vs. Verification

According to the “capability maturity model,” the two terms may be defined as:

- Software Validation: The process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements.
- Software Verification: The process of evaluating software to determine whether the products of a given development

phase satisfy the conditions imposed at the start of that phase (refer Osherove [50]).

In the above definition of software validation, the “specified requirements” though mean ultimate user requirements, there could, however, be requirements from internal customers as well.

Software verification however, may be done in different phases of software development process in order to ensure that the output of each such phase satisfies the requirements, design, etc. of that phase.

In other words, software validation is a process of ensuring that the final software product meets the requirements (needs) of all the stakeholders, though some of these needs might not have been spoken out clearly. This, in a way, also checks whether the documented requirements could capture all the needs of the stakeholders. The oft-quoted saying is that software verification ensures that “you built it right” and confirms that the product, as provided, fulfills the plans of the developers. Software validation ensures that “you built the right thing” and confirms that the product, as provided, fulfills the intended use and goals of the stakeholders.

22.4.14 Conclusion

Each of software testing, validation and verification are essential in a program and all of them must be executed by balancing each other for the increased efficiency of the software. It must be remembered that different error filters are provided by each of them. All the processes are used to find a defect or a bug in a different manner, so the significance of each needs to be considered and utilized accordingly.

While software testing is an activity to investigate software under test in order to provide quality-related information to stakeholders, verification and validation must meet the compliance requirements of law regulated industries, which are often guided by government agencies. The success of software depends upon the acceptance of its targeted audience, easy graphical user interface, and strong functionality load test, among others. Thus, when a software product is being developed, the organization needs to assess whether it will be beneficial to its purchasers as well as other audience.

References

1. Abdeen, M.M.: A model for the fda general principles of software validation (Unpublished doctoral dissertation) (2007)
2. Aggarwala, R.: Progressive interval censoring: some mathematical results with applications to inference. *Commun. Stat. Theory Methods* **30**(8–9), 1921–1935 (2001)
3. Ammann, P., Offutt, J.: *Introduction to Software Testing*. Cambridge University Press (2016)

4. Balakrishnan, N.: Progressive censoring methodology: an appraisal. *Test* **16**(2), 211 (2007)
5. Balakrishnan, N., Balakrishnan, N., Aggarwala, R.: *Progressive Censoring: Theory, Methods, and Applications*. Springer Science and Business Media (2000)
6. Balakrishnan, N., Kannan, N., Lin, C.-T., Ng, H.T.: Point and interval estimation for gaussian distribution, based on progressively type-II censored samples. *IEEE Trans. Reliab.* **52**(1), 90–95 (2003)
7. Balakrishnan, N., Kannan, N., Lin, C., Wu, S.: Inference for the extreme value distribution under progressive type-II censoring. *J. Stat. Comput. Simul.* **74**(1), 25–45 (2004)
8. Balakrishnan, N., Han, D., Iliopoulos, G.: Exact inference for progressively type-I censored exponential failure data. *Metrika* **73**(3), 335–358 (2011)
9. Basin, S.L.: *Measuring the Error Content of Software*. Science Applications, Palo Alto, CA (1974)
10. Brooks, W.D., Motley, R.W.: *Analysis of Discrete Software Reliability Models* Technical Report. Rome Air Development Centre, New York (1980)
11. Brown, J.R., Lipow, M.: *Testing for Software Reliability*. Proc. Intl. Conf. Reliable Software., Los Angeles, California (1975)
12. Chakraborty, A.K.: *Software quality testing and remedies* (Unpublished doctoral dissertation) (1996)
13. Chakraborty, A., Arthanari, T.: Optimum testing time for software under an exploration model. *Opsearch* **31**, 202–214 (1994)
14. Chakraborty, A.K., Basu, P.K., Chakravarty, S.C.: *Guide to ISO 9001: 2000 implementation, improvement tools and techniques and transition to TQM and Six Sigma*. Asian Books Private Ltd., New Delhi (2005)
15. Chakraborty, A.K., Basak, G.K., Das, S.: Bayesian optimum stopping rule for software release. *Opsearch* **56**(1), 242–260 (2019a)
16. Chakraborty, A.K., Soumen, D.: Estimating software reliability using size-biased concepts. [arXiv:2202.081072](https://arxiv.org/abs/2202.081072) (2022)
17. Cho, C.K.: *Quality programming: developing and testing software with statistical quality control*. Wiley (1987)
18. Cohen, A.C.: Progressively censored samples in life testing. *Technometrics* **5**(3), 327–339 (1963)
19. Dalal, S.R., Mallows, C.L.: When should one stop testing software? *J. Am. Stat. Assoc.* **83**(403), 872–879 (1988)
20. Dalal, S.R., Mallows, C.L.: Optimal stopping with exact confidence on remaining defects. *Technometrics* **50**(3), 397–406 (2008)
21. Dewanji, A., Sengupta, D., Chakraborty, A.K.: A discrete time model for software reliability with application to a flight control software. *Appl. Stochast. Models Bus. Ind.* **27**(6), 723–731 (2011)
22. Dhillon, B.S.: *Engineering Maintainability: How to Design for Reliability and Easy Maintenance*. Gulf Professional Publishing (1999)
23. Epstein, B.: Truncated life tests in the exponential case. *Ann. Math. Stat.*, 555–564 (1954)
24. Everett, G.D., McLeod Jr, R.: *Software testing. Testing Across the Entire Software Development Life Cycle*. Wiley (2007)
25. Freund, E.: IEEE standard for system and software verification and validation (IEEE std 1012–2012). *Softw. Qual. Profess.* **15**(1), 43 (2012)
26. Goel, A.L.: Software reliability models: Assumptions, limitations, and applicability. *IEEE Trans. Softw. Eng.* **SE-11**(12), 1411–1423 (1985)
27. Goel, A.L., Okumoto, K.: A Markovian model for reliability and other performance measures of software systems. In: 1979 International Workshop on Managing Requirements Knowledge (Mark), pp. 769–774 (1979a)
28. Goel, A.L., Okumoto, K.: Time-dependent error-detection rate model for software reliability and other performance measures. *IEEE Trans. Reliab.* **28**(3), 206–211 (1979b)
29. Jelinski, Z., Moranda, P.: Software reliability research. In: *Statistical Computer Performance Evaluation*, pp. 465–484. Elsevier (1972)
30. Jorgensen, P.C.: *Software Testing: A Craftsman’s Approach*. Auerbach Publications (2013)
31. Kaner, C.: *Exploratory testing*. In: *Quality Assurance Institute Worldwide Annual Software Testing Conference* (2006)
32. Koopman, P.: *Reliability, safety, and security in everyday embedded systems*. *Lect. Notes Comput. Sci.* **4746**, 1 (2007)
33. Lawless, J.F.: *Statistical Models and Methods for Lifetime Data*, vol. 362. Wiley (2011)
34. Linger, R.C., Mills, H.D., Witt, B.I.: *Structured programming: theory and practice* (1979)
35. Lipow, M.: *Estimation of Software Package Residual Errors*. TRW Software Series TRW-SS-72-09, Redondo Beach, CA (1972)
36. Littlewood, B., Miller, D.R.: Conceptual modeling of coincident failures in multiversion software. *IEEE Trans. Softw. Eng.* **15**(12), 1596–1614 (1989)
37. Littlewood, B., Verrall, J.L.: Likelihood function of a debugging model for computer software reliability. *IEEE Trans. Reliab.* **30**(2), 145–148 (1981)
38. Lyu, M.R., et al.: *Handbook of Software Reliability Engineering*, vol. 222. IEEE Computer Society Press, CA (1996)
39. Martin, R.C.: *Clean Code: A Handbook of Agile Software Craftsmanship*. Pearson Education (2009)
40. Meeker, W.Q., Escobar, L.A.: *Statistical Methods for Reliability Data*. Wiley (2014)
41. Mills, H.D.: On statistical validation of computer programs. IBM Report FSC72-6015, Federal Systems Division, IBM, Gaithersburg, MD (1972)
42. Mills, H., Dyer, M., Linger, R.C.: *Cleanroom software engineering*. *IEEE Software* **1**(9), 19–25 (1987)
43. Musa, J.D.: A theory of software reliability and its application. *IEEE Trans. Softw. Eng.* **SE-1**(3), 312–327 (1975)
44. Musa, J.D., Okumoto, K.: A logarithmic poisson execution time model for software reliability measurement. In: *Proceedings of the 7th International Conference on Software Engineering*, pp. 230–238 (1984)
45. Musa, J., Iannino, A., Okumoto, K.: *Software Reliability: Measurement, Prediction, Application*. McGraw-Hill, New York (1987)
46. Myers, G.J., Sandler, C., Badgett, T.: *The Art of Software Testing*. Wiley, Canada (1979)
47. Nayak, T.K.: Estimating population size by recapture sampling. *Biometrika* **75**(1), 113–120 (1988)
48. Nelson, W.: *Weibull and extreme value distributions*. *Applied Life Data Analysis*. Wiley, New York (1982)
49. Nelson, W.B.: *Accelerated Testing: Statistical Models, Test Plans, and Data Analysis*, vol. 344. Wiley, New York (2009)
50. Osherove, R.: *The Art of Unit Testing: With Examples in .net*. Manning Publications Co. (2009)
51. Pham, H.: *Software Reliability*. Wiley Encyclopedia of Electrical and Electronics Engineering (2001)
52. Pham, H.: *Handbook of Reliability Engineering*. Springer Science and Business Media (2006)
53. Pradhan, B., Kundu, D.: On progressively censored generalized exponential distribution. *Test* **18**(3), 497 (2009)
54. Ramamoorthy, C.V., Bastani, F.B.: Software reliability status and perspectives. *IEEE Trans. Softw. Eng.* **SE-8**(4), 354–371 (1982)
55. Roy, S., Gijo, E., Pradhan, B.: Inference based on progressive type I interval censored data from log-normal distribution. *Commun. Stat. Simul. Comput.* **46**(8), 6495–6512 (2017)
56. Schick, G.J., Wolverton, R.W.: An analysis of competing software reliability models. *IEEE Trans. Softw. Eng.* **SE-4**(2), 104–120 (1978)

57. Schneidewind, N.F.: Software reliability model with optimal selection of failure data. *IEEE Trans. Softw. Eng.* **19**, 1095–1104 (1993)
58. Singpurwalla, N.D.: Determining an optimal time interval for testing and debugging software. *IEEE Trans. Softw. Eng.* **17**(4), 313–319 (1991)
59. Standard, M.: 781-c, Reliability Design Qualification and Production Acceptance Tests, Exponential Distribution. Department of Defense, Washington, DC (1977)
60. U.S. Department Of Health and Human Services.: General Principles of Software Validation; Final Guidance for Industry and FDA Staff, Washington, DC (2002)
61. Vasanthi, T., Arulmozhi, G.: Software reliability estimation using bayesian approach. *Int. J. Qual. Reliabil. Manag.* (2013)
62. Viveros, R., Balakrishnan, N.: Interval estimation of parameters of life from progressively censored data. *Technometrics* **36**(1), 84–91 (1994)
63. Yang, B., Xie, M.: A study of operational and testing reliability in software reliability analysis. *Reliabil. Eng. Syst. Safe* **70**(3), 323–329 (2000)



Dr. Ashis Kumar Chakraborty is a senior faculty in the Statistical Quality Control and Operations Research (SQC & OR) Division of Indian Statistical Institute (ISI), Kolkata. He has published four books and contributed in many other books. His research publications include the areas of Software and Hardware reliability, Process control, Data Analytics etc. He is a life member of the Operations Research Society of India and also of the Indian Association for productivity, Quality and Reliability. He was elected as Head of the SQC & OR Division, ISI, twice and also served as the leader of Quality Mission under aegis of Government of India.



Dr. E. V. Gijo is a Faculty at Statistical Quality Control and Operations Research unit of Indian Statistical Institute at Bangalore. He has authored a book on Lean Six Sigma and published around 45 articles in international journals related Six Sigma, Statistics, Reliability, Quality

Management and Time Series Forecasting. Dr. Gijo has worked on a number of consultancy projects with quite a few organizations.



Anisha Das is working as a research scholar in Biostatistics at the Florida State University. She was a visiting scholar at the Indian Statistical Institute where she handled a few projects. Her research interests include Bayesian analysis, bootstrapping and artificial intelligence techniques. She is well proficient in R and SAS programming and currently a member of American Association for the Advancement of Science.



Moutushi Chatterjee received her M. Sc. in Statistics from Kalyani University and M. Tech (Quality, Reliability and Operations Research) and PhD from Indian Statistical Institute, Kolkata, India. She is an Assistant Professor of Statistics at Lady Brabourne College (affiliated to University of Calcutta). Her current research interests include statistical quality control, multivariate statistics, reliability and operations research.



OSS Reliability Analysis and Project Effort Estimation Based on Computational Intelligence

23

Shigeru Yamada, Yoshinobu Tamura , and Kodai Sugisaki

Contents

23.1	Introduction	443
23.2	Reliability of OSS	444
23.2.1	Open-Source Project and Reliability.....	444
23.2.2	OSS and Definitions of Reliability.....	445
23.2.3	Development Paradigm of OSS.....	445
23.3	Stochastic Modeling Approach	446
23.3.1	Hazard Rate Model.....	446
23.3.2	NHPP Model.....	447
23.3.3	Stochastic Differential Equation Modeling for OSS.....	448
23.4	Pixel Data-Driven Approach Based on Software GUI	448
23.4.1	Effort Estimation Model.....	448
23.4.2	Parameter Estimation of Maintenance Effort Model Based on the Deep Learning.....	449
23.4.3	The Procedure of Our Method.....	449
23.4.4	Goodness-of-Fit Comparison.....	450
23.4.5	Numerical Illustrations.....	450
23.5	Calculation by MXNet Package	451
23.6	Concluding Remarks	453
	References	454

quality for the progress status of OSS project, because the software failure is caused by the poor handling of effort control. In particular, the GUI of OSS will be frequently made a dramatic difference according to the major version upgrade. The changing in GUI of OSS will depend on the development and management effort of OSS in the specified version. Considering the relationship between GUI and OSS development process, the UX/UI design of OSS will change with the procedure of OSS development. This chapter focuses on the method of effort estimation for OSS project. Then, the pixel data and OSS fault big data are analyzed by using the deep learning. Moreover, we discuss the effort assessment method in the development phase by using the effort data.

Keywords

Open source software · Stochastic model · Deep learning · GUI

Abstract

OSS (open-source software) systems serve as the key components of critical infrastructures in the society. As for the OSS development paradigm, the bug tracking systems are used for software quality management in many OSS projects. It is important to appropriately control the

23.1 Introduction

Many traditional software reliability growth models [1–3] (SRGMs) have been used to assess the reliability for quality management under the system testing phase of waterfall software development. Moreover, we have been energetically proposed several reliability analysis methods based on computational intelligence for an open-source software (OSS) in the last decade; see e.g., [4]. Its measurement and management technologies for OSS are essential to produce and maintain reliable system for OSS. As for the OSS development paradigm, the bug tracking systems are used for software quality management in many OSS projects. In the bug tracking systems, many fault data sets are recorded by several users and developers. It will be helpful for OSS project managers to assess the reliability and maintainability of OSS, if many fault data recorded on the bug tracking

S. Yamada (✉)
Tottori University, Tottori, Japan
e-mail: yamada@tottori-u.ac.jp

Y. Tamura
Yamaguchi University, Yamaguchi, Japan
e-mail: tamuray@yamaguchi-u.ac.jp

K. Sugisaki
Tokyo City University, Tokyo, Japan
e-mail: g1981826@tcu.ac.jp

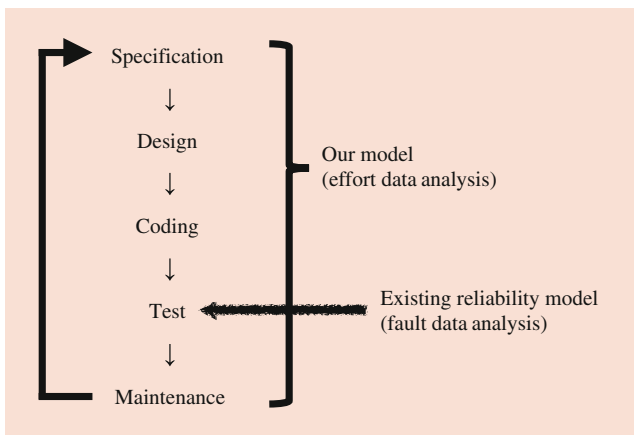


Fig. 23.1 The fault and effort data analysis

system are used for software quality improvement. Further, it is efficient to compare with the conventional methods [1–3] based on SRGMs if the software managers can use the development effort data in bug tracking system in terms of QCD (quality, cost, delivery) [4].

In this chapter, we focus on the method of effort estimation based on the deep learning for OSS project by using the OSS fault big data [5–7]. For example, Fig. 23.1 represents the relationship of fault and effort data analysis in the case of the Waterfall model. It is important to appropriately manage the quality according to the progress status of OSS project, because the software failure is caused by the poor handling of effort control as shown in Fig. 23.1. In particular, we can totally assess the development phase by using the effort data. Also, the appropriate control of management effort for OSS will indirectly link to the quality, reliability, and cost. Also, the GUI of OSS will be frequently made a dramatic difference according to the major version-upgrade.

Then, we discuss the method of effort estimation of OSS project based on the deep learning. In particular, we use the GUI data and the effort data obtained from fault big data. Then, we use the pixel data sets obtained from GUI of OSS for the actual OSS project. Also, several numerical examples of the proposed method based on the deep learning by using the pixel data and maintenance effort data in the actual OSS projects are shown in this chapter. Moreover, we discuss the results of numerical examples by using our proposed method.

23.2 Reliability of OSS

23.2.1 Open-Source Project and Reliability

All over the world, people can obtain a lot of information at the same time by growing rate of Internet access around the world in recent years. In accordance with such a pene-

tration of the Internet, it is increasing public awareness of the importance of online real-time and interactive functions. Therefore, the distributed software development paradigm based on the information technologies are expanding at an explosive pace. Especially, new development paradigm, such as client/server system, web programming, object-oriented development, and also on, by using network technologies have been in heavy usage by the software developers.

At present, there are a number of development paradigms within the same company, the development environment joined hands with several software company, and the collaborative development based on consortia formation in terms of the distributed development environment. Furthermore, a software development paradigm based on an open-source project is rapidly spreading.

The open-source project contains special features so-called software composition. These geographically dispersed several components are developed in all parts of the world. The successful experiences as the distributed development model in open-source projects include GNU/Linux operating system,¹ Apache Web server, and so on.² Especially, OSS is frequently applied as server use, instead of client use. Then, such an open-source system development is still ever-expanding now.

A computer-software is developed by human work; therefore, many software faults must be introduced into the software product during the development process. Thus, these software faults often cause complicated breakdowns of computer systems. Recently, it becomes more difficult for the developers to produce highly reliable software systems efficiently because of the diversified and complicated software requirements. Therefore, it is necessary to control the software development process in terms of quality and reliability. Note that a *software failure* is defined as an unacceptable departure of program operation caused by a *software fault* remaining in the software system. Basically, the software reliability can be evaluated by the number of detected faults or the software failure-occurrence time. Especially, the *software reliability models* which can describe software fault-detection or failure-occurrence phenomena are called SRGMs [1]. The SRGMs are useful to assess the reliability for quality control and testing-process control of software development.

On the other hand, the effective testing management method for distributed development environment as typified by open-source project has only a few presented [8, 9]. In the case of considering the effect of a debugging process on entire system in the development of a method of reliability assessment for the distributed development environment, it is necessary to grasp the deeply intertwined factors, such as

¹Linux is a Registered Trademark of Linus Torvalds.

²Other company, product, or service names may be trademarks or service marks of others.

programming path, size of component, skill of fault reporter, and so on.

23.2.2 OSS and Definitions of Reliability

At present, OSS systems serve as the key components of critical infrastructures in the society. OSS projects possess a unique feature known as software composition by which components are developed by teams that are geographically dispersed throughout the world. Successful OSS projects include Apache HTTP server, MySQL database server, OpenStack cloud software, Firefox Web browser, and GNU/Linux operating system. However, poor handling of quality issues and customer support has limited the progress of OSS, because the development cycle of OSS has no testing phase. For example, mobile OSS has been gaining a lot of attention in the embedded system area, i.e., Android, BusyBox, and Firefox OS. However, the installer software developed under the third-party developers indirectly affects the reliability of a mobile device. Therefore, it is difficult for many companies to assess the reliability of a mobile OSS, because a mobile OSS includes several software versions. Another closely related issue is that of software vulnerability posed by the open-source nature of the code, raising the possibility of security loopholes. For the abovementioned reasons, it is difficult for software managers to assess the reliability of OSS.

We compare the characteristics of software reliability with those of hardware reliability as follows:

Hardware Reliability

1. Hardware failures can be due to wear out.
2. Hardware reliability is affected by deficiencies injected during all phases of the development, operation, and maintenance.
3. Hardware reliability can be improved by redundancy with identical units.
4. A testing method of hardware reliability is established and standardized.
5. A maintenance technology is advanced since the market of hardware products is established and the user environment is seized.

Software Reliability

1. Software failures can be due to no wear-out phenomenon.
2. Software reliability is, inherently, determined during the earlier phase of the development process, i.e., specification and design phases.
3. Software reliability cannot be improved by redundancy with identical versions.

4. A verification method of software reliability has not been established.
5. A maintenance technology is not established since the market of software products is rather recent.

Generally, software failures caused by software faults latent in the software system cannot occur expected for a special occasion when a set of special data is put into the system under a special condition, i.e., the program path including software faults is executed. Therefore, the software reliability is determined by the input data and the internal condition of the program. We summarize the definitions of the terms related to the software reliability in the following.

A software system is a product which consists of the programs and documents as a result of the software development process. Specification derived by analyzing user requirements for the software system is a document which describes the expected performance of the system. When the software performance deviates from the specification and the output variable has an improper value or the normal processing is interrupted, it is said that a software failure occurs. That is, *software failure* is defined as a departure of program operation from the program requirements. The cause of software failure is called a software fault. Then, *software fault* is defined as a defect in the program which causes a software failure. The software fault is usually called software bug. *Software error* is defined as human action that results in the software system containing a software fault. Thus, the software fault is considered to be a manifestation of software errors. In this chapter, software error may be used to mean a defect detected and corrected during the testing or the operation phase without distinction from software fault.

23.2.3 Development Paradigm of OSS

At present, many OSS are developed under several open-source projects. For example, the Firefox Web browser, Firefox OS, and Thunderbird mailer are developed and managed under the Mozilla.org project. Also, Apache HTTP server, Tomcat, and Flex are developed and managed under the Apache software foundation. These open-source projects control the development phase of many open source projects. The open-source project has no the specific testing phase. Also, the specification of OSS continuously changes with the version upgrade, because the software of several versions is uploaded to the website of OSS project.

It is difficult for software managers to assess OSS reliability because of the differences among the development style of OSS and traditional software. Therefore, it is important to assess and manage considering the characteristics of the OSS development paradigm.

In particular, OSS have several versions of the development process as follows:

1. Bug Fix Version (most urgent issue such as patch)
2. Minor Version (minor revision by the addition of a component and module)
3. Major Version (significant revision for specification)

Also, the version number of OSS is generally described as the “(Major version number. Minor version number. Revision number. Build number),” e.g., (2.1.2103.1104). There are several versions for each OSS. Therefore, it is known that it is difficult for software managers to select the appropriate OSS, because several OSS are uploaded to the website of open-source project.

This chapter focuses on the reliability of OSS with the abovementioned characteristics. Several methods of reliability assessment for OSS are presented in this chapter. Also, several numerical examples for each method of reliability assessment based on computational intelligence are given by using actual fault data of OSS. These methods introduced in this chapter may be useful for software managers to assess the reliability of a software system developed using the OSS paradigm.

23.3 Stochastic Modeling Approach

Generally, a mathematical model based on probability theory and statistics is useful to describe the software fault-detection phenomena or the software failure-occurrence phenomena and estimate the software reliability quantitatively. During the testing phase in the software development process, software faults are detected and removed with a lot of test-effort expenditures. Then, the number of remaining faults in the software system is decreasing as the testing goes on. This means that the probability of software failure occurrence is decreasing so the software reliability is increasing and the time-interval between software failures becomes longer with the testing time. A mathematical tool which treats such software reliability aspect is a SRGM [1].

23.3.1 Hazard Rate Model

Based on the definitions discussed previous section, we can make a software reliability growth model based on the assumptions for actual environments during the testing phase or the operation phase. Then, we can define the following random variables on the number of detected faults and the software failure-occurrence time.

- $N(t)$: the cumulative number of detected software faults (or the cumulative number of observed software failures), up to testing time t ,
- S_i : the i -th software failure occurrence time ($i = 1, 2, \dots$; $S_0 = 0$),
- X_i : the time-interval between $(i - 1)$ -st and i -th software failures ($i = 1, 2, \dots$; $X_0 = 0$).

Figure 23.2 shows the occurrence of event $\{N(t) = i\}$ since i th faults have been detected up to time t . From these definitions, we have

$$S_i = \sum_{k=1}^i X_k, \quad X_i = S_i - S_{i-1}. \quad (23.1)$$

Assuming that the hazard rate, i.e., the *software failure rate*, for X_i ($i = 1, 2, \dots$), $z_i(x)$, is proportional to the current number of residual faults remaining in the system, we have

$$z_i(x) = (N - i + 1)\lambda(x), \quad i = 1, 2, \dots, N; x \geq 0, \quad \lambda(x) > 0, \quad (23.2)$$

where N is the initial fault content and $\lambda(x)$ the software failure rate per fault remaining in the system at time x . If we consider two special cases in Eq. (23.2) as:

$$\lambda(x) = \phi, \quad \phi > 0, \quad (23.3)$$

$$\lambda(x) = \phi x^{m-1}, \quad \phi > 0, m > 0, \quad (23.4)$$

then two typical software hazard rate models, respectively, called the Jelinski-Moranda model and the Wagoner model can be derived, where ϕ and m are constant parameters. Usually, it is not completely fault free or failure free. Then, we have a software hazard rate model called the Moranda model for the case of the infinite number of software failure occurrences as

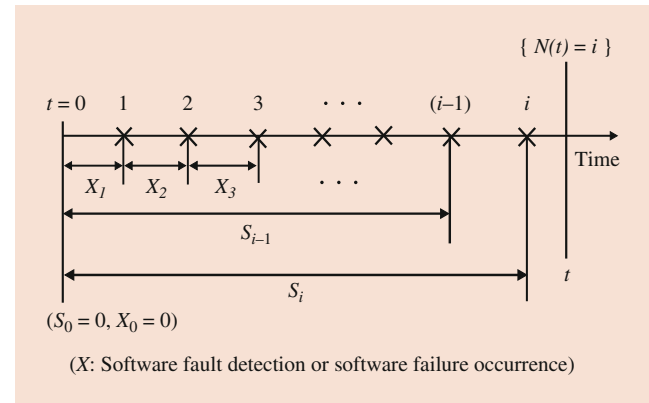


Fig. 23.2 The stochastic quantities related to a software fault detection phenomenon or a software failure occurrence phenomenon

$$z_i(x) = Dk^{i-1}, \quad i = 1, 2, \dots; D > 0, 0 < k < 1, \quad (23.5)$$

where D is the initial software hazard rate and k the decreasing ratio. Equation (23.5) describes a software failure-occurrence phenomenon where a software system has high frequency of software failure occurrence during the early stage of the testing or the operation phase and it gradually decreases thereafter. Based on the software hazard rate models above, we can derive several software reliability assessment measures. For example, the software reliability function for X_i ($i = 1, 2, \dots$) is given as

$$R_i(x) = \exp\left[-\int_0^x z_i(x)dx\right], \quad i = 1, 2, \dots. \quad (23.6)$$

23.3.2 NHPP Model

Further, we also discuss nonhomogeneous Poisson process (NHPP) models, which are modeled for random variable $N(t)$ as typical SRGMs. In the NHPP models, an NHPP is assumed for the random variable $N(t)$, the distribution function of which is given by

$$\Pr\{N(t) = n\} = \frac{H(t)^n}{n!} \exp[-H(t)], \quad n = 1, 2, \dots, \\ H(t) \equiv E[N(t)] = \int_0^t h(x)dx, \quad (23.7)$$

where $\Pr[\cdot]$ and $E[\cdot]$ mean the probability and expectation, respectively. $H(t)$ in Eq. (23.7) is called a mean value function which indicates the expectation of $N(t)$, i.e., the expected cumulative number of faults detected (or the expected cumulative number of software failures occurred) in the time interval $(0, t]$, and $h(t)$ in Eq. (23.7) called an intensity function which indicates the instantaneous fault-detection rate at time t . From Eq. (23.7), various software reliability assessment measures can be derived. For example, the expected number of faults remaining in the system at time t is given by:

$$n(t) = a - H(t), \quad (23.8)$$

where $a \equiv H(\infty)$, i.e., parameter a denotes the expected initial fault content in the software system. Given that the testing or the operation has been going on up to time t , the probability that a software failure does not occur in the time interval $(t, t+x](x \geq 0)$ is given by conditional probability $\Pr\{X_i > x | S_{i-1} = t\}$ as:

$$R(x|t) = \exp[H(t) - H(x+t)], \quad t \geq 0, x \geq 0. \quad (23.9)$$

$R(x|t)$ in Eq. (23.9) is a so-called *software reliability*. Measures of MTBF (mean time between software failures or fault-detections) can be obtained follows:

$$MTBF_i(t) = \frac{1}{h(t)}, \quad (23.10)$$

$$MTBF_c(t) = \frac{t}{H(t)}. \quad (23.11)$$

MTBFs in Eqs. (23.10) and (23.11) are called *instantaneous MTBF* and *cumulative MTBF*, respectively. It is obvious that the lower the value of $n(t)$ in Eq. (23.8), the higher the value $R(x|t)$ for specified x in Eq. (23.9), or the longer the value of MTBFs in Eqs. (23.10) and (23.11), the higher the achieved software reliability is. Then, analyzing actual test data with accepted NHPP models, these measures can be utilized to assess software reliability during the testing or operation phase, where statistical inferences, i.e., parameter estimation and goodness-of-fit test, are usually performed by a method of maximum likelihood.

To assess the software reliability actually, it is necessary to specify the mean value function $H(t)$ in Eq. (23.7). Many NHPP models [1] considering the various testing or operation environments for software reliability assessment have been proposed in the last decade. As discussed above, a software reliability growth is described as the relationship between the elapsed testing or operation time and the cumulative number of detected faults and can be shown as the reliability growth curve mathematically.

Among the NHPP models, the exponential and the modified exponential software reliability growth models are appropriate when the observed reliability growth curve shows an exponential curve. Similarly, the delayed S-shaped and the inflection S-shaped software reliability growth models are appropriate when the reliability growth curve is S-shaped.

In addition, as for computer makers or software houses in Japan, logistic curve and Gompertz curve models have often been used as software quality assessment models, on the assumption that software fault-detection phenomena can be shown by S-shaped reliability growth curves. In these deterministic models, the cumulative number of faults detected up to testing t is formulated by the following growth equations:

$$L(t) = \frac{k}{1+me^{-at}}, \quad m > 0, \alpha > 0, k > 0, \quad (23.12)$$

$$G(t) = ka^{(bt)}, \quad 0 < a < 1, 0 < b < 1, k > 0. \quad (23.13)$$

In Eqs. (23.12) and (23.13), assuming that a convergence value of each curve ($L(\infty)$ or $G(\infty)$), i.e., parameter k , represents the initial fault content in the software system, it can be estimated by a regression analysis.

23.3.3 Stochastic Differential Equation Modeling for OSS

We focus on an OSS developed under the open-source project. We discuss a useful software reliability assessment method in open-source project as a typical case of new distributed development paradigm. Especially, we introduce a software reliability growth model based on stochastic differential equations in order to consider the active state of the open source project. We assume that the software failure intensity depends on the time, and the software fault-report phenomena on the bug tracking system keep an irregular state.

Let $S(t)$ be the number of faults in the OSS system at testing time t ($t \geq 0$). Suppose that $S(t)$ takes on continuous real values. Since latent faults in the OSS system are detected and eliminated during the operational phase, $S(t)$ gradually increases as the operational procedures go on. Thus, under common assumptions for the software reliability growth modeling, we consider the following linear differential equation:

$$\frac{dS(t)}{dt} = \lambda(t)S(t), \quad (23.14)$$

where $\lambda(t)$ is the intensity of inherent software failures at operational time t and is a nonnegative function.

In most cases, the faults of OSS are not reported to the bug tracking system at the same time as fault detect but rather reported to the bug tracking system with the time lag of fault-detection and report. As for the fault report to the bug tracking system, we consider that the software fault-report phenomena on the bug tracking system keep an irregular state. Moreover, the addition and deletion of software component is repeated under the development of OSS, i.e., we consider that the software failure intensity depends on the time.

Therefore, we suppose that $\lambda(t)$ in Eq. (23.14) has the irregular fluctuation. That is, we extend Eq. (23.14) to the following stochastic differential equation [4]:

$$\frac{dS(t)}{dt} = \{\lambda(t) + \sigma\gamma(t)\} S(t), \quad (23.15)$$

where σ is a positive constant representing a magnitude of the irregular fluctuation and $\gamma(t)$ a standardized Gaussian white noise.

We extend Eq. (23.15) to the following stochastic differential equation of an Itô type:

$$dS(t) = \left\{ \lambda(t) + \frac{1}{2}\sigma^2 \right\} S(t)dt + \sigma S(t)dW(t), \quad (23.16)$$

where $W(t)$ is a one-dimensional Wiener process which is formally defined as an integration of the white noise $\gamma(t)$ with

respect to time t . The Wiener process is a Gaussian process and it has the following properties:

$$\Pr[W(0) = 0] = 1, \quad (23.17)$$

$$E[W(t)] = 0, \quad (23.18)$$

$$E[W(t)W(t')] = \text{Min}[t, t']. \quad (23.19)$$

By using Itô's formula [4], we can obtain the solution of Eq. (23.15) under the initial condition $S(0) = v$ as follows:

$$S(t) = v \cdot \exp\left(\int_0^t \lambda(s)ds + \sigma W(t)\right), \quad (23.20)$$

where v is the number of detected faults for the previous software version. Using solution process $S(t)$ in Eq. (23.20), we can derive several software reliability measures.

Moreover, we define the intensity of inherent software failures, $\lambda(t)$, as follows:

$$\int_0^t \lambda(s)ds = (1 - \exp[-\alpha t]), \quad (23.21)$$

where α is an acceleration parameter of the intensity of inherent software failures.

From Eq. (23.20), we can confirm that the number of detected faults cannot converge to a finite value as the following equation:

$$\lim_{t \rightarrow \infty} S(t) = \infty. \quad (23.22)$$

The operation environment of OSS has the characteristics of the susceptible to various operational environments. Therefore, it is different from conventional software systems developed under the identical organization. Then, the expected number of detected faults continues to increase from the effect of the interaction among various operational environments, i.e., the number of detected faults can not converge to a finite value [4].

23.4 Pixel Data-Driven Approach Based on Software GUI

23.4.1 Effort Estimation Model

Considering the characteristic of the operation phase of OSS projects, the time-dependent expenditure phenomena of maintenance effort depend on the change in UI/UX design during the operation phase, because the OSS are developed and maintained by several developers and users. UI means the user interface, UX the user experience. The operation phases of many OSS projects are influenced from external

factors by triggers such as the difference of skill, time lag of development, and maintenance. Considering the above points, we apply the mathematical model to manage the OSS project. Then, let $\lambda(t)$ be the cumulative maintenance effort expenditures up to operational time $t(t \geq 0)$ in the OSS development project. Suppose that $\lambda(t)$ takes on continuous real values. Since the estimated maintenance effort can be observed during the operational phase of the OSS project, $\lambda(t)$ gradually increases as the operational procedures go on. Based on the same approach in software reliability growth modeling [1–3], the following linear differential equation in terms of maintenance effort can be formulated:

$$\frac{d\Lambda(t)}{dt} = \beta \{\alpha - \Lambda(t)\}, \tag{23.23}$$

where β is the increase rate of maintenance effort expenditures at operational time t , and α means the estimated amount of maintenance effort required until the end of operation. Then, we can solve Eq. (23.23) under the initial condition $\lambda(0) = 0$. Therefore, the cumulative maintenance effort expenditures up to time t are obtained as follows:

$$\Lambda(t) = \alpha(1 - e^{-\beta t}). \tag{23.24}$$

Moreover, we discuss the parameter estimation by considering the UX/UI design, because UX/UI design changes with the operation procedure, i.e., we consider that the changing in UX/UI design depends on the cumulative maintenance effort expenditures. We assume that a set of data $(t_i, \lambda_i)(i = 1, 2, \dots, n)$ in terms of development effort data is obtained from the bug tracking system, where λ_i is the cumulative maintenance effort expenditures up to operational time t_i in the OSS development project. Then, we consider the following two kinds of data $o_1 (0 < o_1 < 1)$ and $o_2 (0 < o_2 < 1)$ as the learning data in the deep learning:

$$o_1 = \frac{\lambda_n - \lambda_b}{\lambda_n}, \tag{23.25}$$

where λ_b is the cumulative maintenance effort expenditures up to previous version. Then, the parameter α in Eq. (23.24) is given by $\alpha \equiv \lambda_n (1 + o_1)$. From Eq. (23.24), we can obtain the following equation:

$$\beta \equiv o_2 = \frac{1}{t_n} \ln \left[\frac{\alpha}{\alpha - \lambda_n} \right]. \tag{23.26}$$

Abovementioned $o_1 (0 < o_1 < 1)$ and $o_2 (0 < o_2 < 1)$ are obtained from the actual data on the bug tracking system are used as the learning data.

23.4.2 Parameter Estimation of Maintenance Effort Model Based on the Deep Learning

In this chapter, we apply the deep feedforward neural network (DFNN) and deep convolutional neural network (DCNN) to learn the UX/UI design of open-source projects. In particular, the GUI of OSS will be frequently made a dramatic difference according to the major version upgrade. Then, it is very important to consider the pixel data such as GUI. We apply the following pixel data of UX/UI design to the parameters of pre-training units. Then, the objective variable is given as the numerical value from 0 to 1 in terms of $o_1 (0 < o_1 < 1)$ and $o_2 (0 < o_2 < 1)$, respectively. In other words, we apply two kinds of parameters $o_1 (0 < o_1 < 1)$ and $o_2 (0 < o_2 < 1)$ to the amount of compressed characteristics, respectively.

The structure of the DFNN in this chapter is shown in Fig. 23.3. In Fig. 23.3, $z_l (l = 1, 2, \dots, L)$ and $z_m (m = 1, 2, \dots, M)$ means the pre-training units. Also, $o_n (n = 1, 2, \dots, N)$ is the amount of compressed characteristics. Several algorithms in terms of the deep learning have been proposed [10–15]. In particular, the network structure of DCNN is shown in Fig. 23.4.

23.4.3 The Procedure of Our Method

The procedures of the proposed method based on the deep learning are shown as follows:

- We obtain the fault big data sets from the bug tracking system.
- The development effort is calculated from the fault big data.

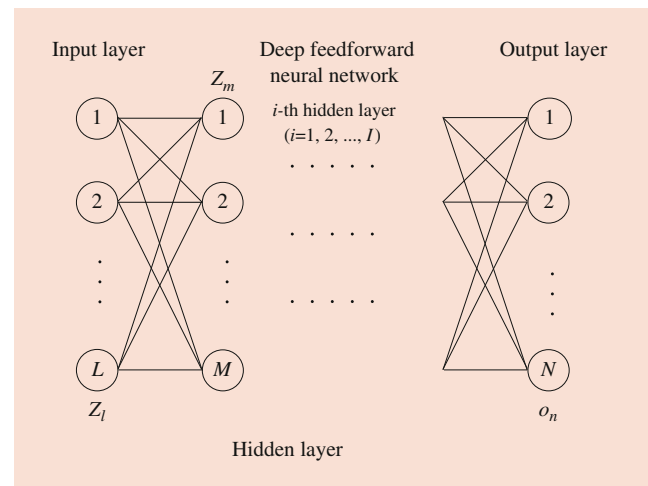


Fig. 23.3 The structure of DFNN

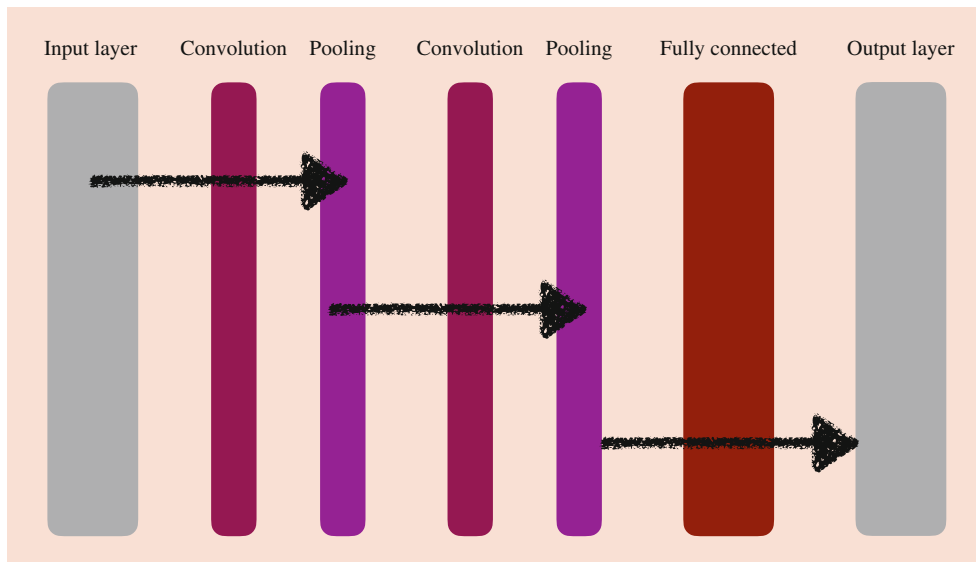


Fig. 23.4 The structure of DCNN

- We collect GUI data as UX/UI design for each software version.
- Then, the pixel data sets of GUI for each software version are given by the standardization of size in GUI.
- The pixel data is used as the input data of the deep learning. Then, the output data is o_1 ($0 < o_1 < 1$) and o_2 ($0 < o_2 < 1$).
- The DFNN and DCNN are used as the deep learning algorithm in this chapter.

23.4.4 Goodness-of-Fit Comparison

We compare the goodness-of-fit of DFNN with DCNN for the observed data set. We adopt the value of the root mean square error (RMSE) as comparison criteria of goodness of fit. Suppose that K data pairs (t_k, y_k) ($k = 1, 2, \dots, K$) are observed during the OSS operating-phase where y_k is the cumulative software effort expenditures observed in the time interval $(0, t_k]$. RMSE can be obtained by using the sum of square errors between the observed value, y_k , and the estimated one, \hat{y}_k , by the number of data pairs, K . That is:

$$\text{RMSE} = \sqrt{\frac{1}{K} \sum_{k=1}^K (y_k - \hat{y}_k)^2}, \quad (23.27)$$

where \hat{y}_k in Eq. (23.27) is obtained from estimated $\hat{\Lambda}(t_k)$ ($k = 1, 2, \dots, K$) in Eq. (23.24). RMSE indicates that the selected model fits better to the observed data as the RMSE becomes small.

23.4.5 Numerical Illustrations

In this chapter, we focus on Apache OpenOffice [16] in order to evaluate the performance of our methods. Several numerical examples by using the data sets for Apache OpenOffice as OSS are shown in this chapter. The data used in this chapter are collected in the bug tracking system on the website of Apache OpenOffice open-source project. We obtain 10,000 fault data sets from the data recorded on bug tracking system of Apache OpenOffice. The development effort data are analyzed by using the obtained 10,000 fault data sets. We show the actual data in Table 23.1.

We show the estimated cumulative maintenance effort expenditures in the case of DFNN and DCNN up to time t in Figs. 23.5 and 23.6. From Figs. 23.5 and 23.6, we find that our method will be helpful for the OSS managers to confirm the trend of maintenance effort by using UX/UI data such as GUI in terms of the long-term prediction. For example, the estimated cumulative software effort expenditures is about 1.20×10^9 man-month in 200 months in Fig. 23.5. The OSS managers can estimate the maintenance effort from the pixel data of GUI by using the proposed method.

Based on the results of DFNN and DCNN, the estimated RMSE in the case of o_1 and o_2 are shown in Figs. 23.7, 23.8, 23.9, 23.10. Moreover, Table 23.2 shows the comparison of goodness of fit based on RMSE in terms of maintenance effort. From Table 23.2, we have found that the estimated RMSE in the case of DCNN fits better than one of DFNN.

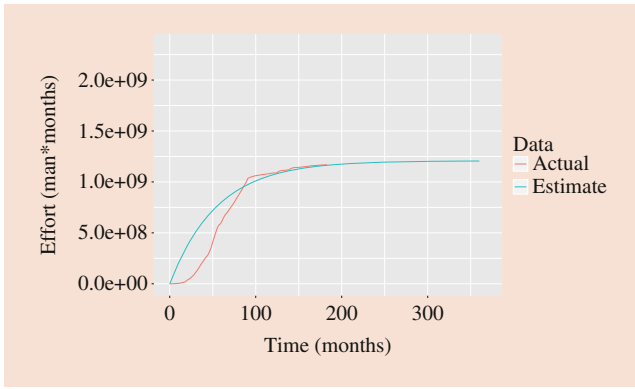


Fig. 23.5 The estimated cumulative software effort in the case of DFNN

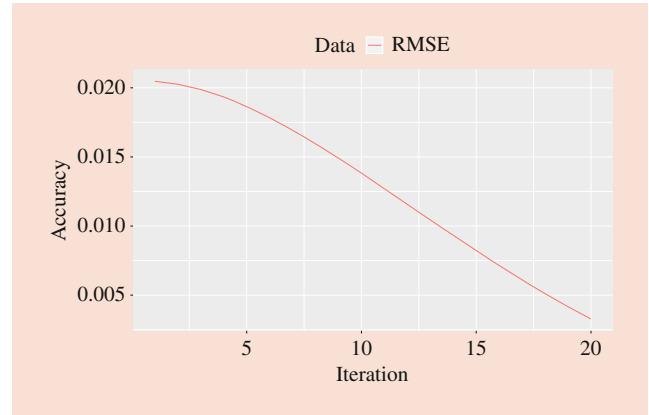


Fig. 23.8 The estimated RMSE of o_2 in the case of DFNN

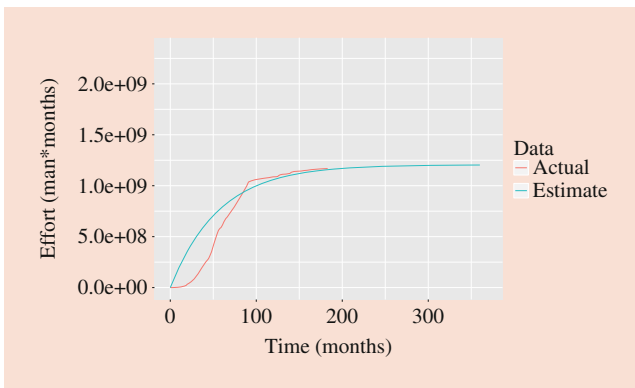


Fig. 23.6 The estimated cumulative software effort in the case of DCNN

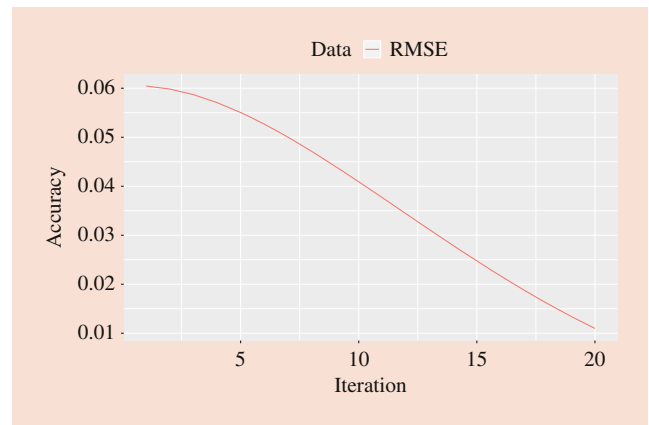


Fig. 23.9 The estimated RMSE of o_1 in the case of DCNN



Fig. 23.7 The estimated RMSE of o_1 in the case of DFNN

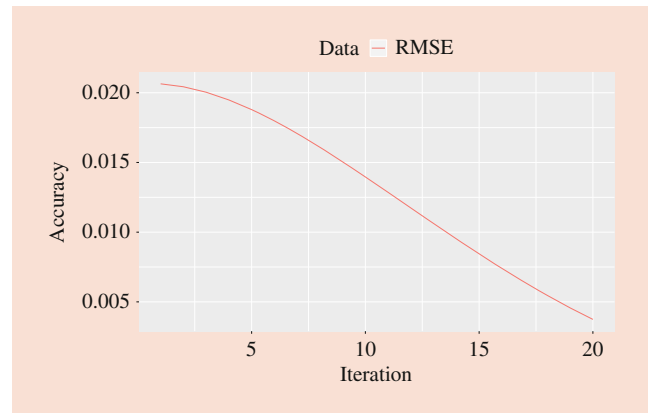


Fig. 23.10 The estimated RMSE of o_2 in the case of DCNN

23.5 Calculation by MXNet Package

In the case of DFNN, we can calculate by using the MXNet package via Statistical Computing R. Then, we can implement the program of DFNN as follows:

Similarly, we can calculate by using the MXNet package via Statistical Computing R in the case of DCNN. Then, we can implement the program of DCNN as follows:

Listing 23.1 Implementation example of DFNN

```
library(Metrics)
library(ggplot2)
library(data.table)

data <- fread("data.csv")

train_data <- head(data.frame(data), n=300)
test_data <- tail(data.frame(data), n=200)

dim(train_data)
dim(test_data)

xx <- data.matrix(train_data[,3:94502])
yy <- train_data[,1]
pxx <- data.matrix(test_data[,3:94502])
pyy <- test_data[,1]

library(mxnet)

data <- mx.symbol.Variable("data")

fc1 <- mx.symbol.FullyConnected(data, name = "fc1", num_hidden = 200)
act1 <- mx.symbol.Activation(fc1, name = "relu1", act_type = "relu")
drop1 <- mx.symbol.Dropout(act1, p = 0.2)

fc2 <- mx.symbol.FullyConnected(drop1, name = "fc2", num_hidden = 150)
act2 <- mx.symbol.Activation(fc2, name = "relu2", act_type = "relu")
drop2 <- mx.symbol.Dropout(act2, p = 0.2)

fc3 <- mx.symbol.FullyConnected(drop2, name = "fc3", num_hidden = 50)
act3 <- mx.symbol.Activation(fc3, name = "relu3", act_type = "relu")
drop3 <- mx.symbol.Dropout(act3, p = 0.2)

fc4 <- mx.symbol.FullyConnected(drop3, num_hidden = 1)

output <- mx.symbol.LinearRegressionOutput(fc4)

mx.set.seed(1)

logger <- mx.metric.logger$new()

model <- mx.model.FeedForward.create(output, X = xx, y = yy,
  ctx=mx.cpu(), num.round=20, array.batch.size=100, learning.rate=0.01,
  momentum=0.9, wd=0.01, eval.metric=mx.metric.rmse,
  epoch.end.callback = mx.callback.log.train.metric(5, logger))
```

Listing 23.2 Implementation example of DCNN

```

library(Metrics)
library(ggplot2)
library(data.table)

data <- fread("data.csv")

train_data <- head(data.frame(data), n=300)
test_data <- tail(data.frame(data), n=200)

dim(train_data)
dim(test_data)

xx <- data.matrix(train_data[,3:94502])
yy <- train_data[,1]
pxx <- data.matrix(test_data[,3:94502])
pyy <- test_data[,1]

library(mxnet)

data <- mx.symbol.Variable("data")

#fc1 <- mx.symbol.FullyConnected(data, name = "fc1", num_hidden = 200)
#act1 <- mx.symbol.Activation(fc1, name = "relu", act_type = "relu")
#drop1 <- mx.symbol.Dropout(act1, p = 0.2)

fc1 <- mx.symbol.Convolution(data=data, kernel=c(5,5), num_filter=10)
act1 <- mx.symbol.Activation(data=fc1, act_type="tanh")
pool1 <- mx.symbol.Pooling(data=act1, pool_type="max", kernel=c(5,5),
stride=c(2,2))

fc2 <- mx.symbol.Convolution(data=pool1, kernel=c(5,5), num_filter=10)
act2 <- mx.symbol.Activation(data=fc2, act_type="tanh")
pool2 <- mx.symbol.Pooling(data=act2, pool_type="max", kernel=c(2,2),
stride=c(2,2))

fc3 <- mx.symbol.FullyConnected(pool2, name = "fc3", num_hidden = 100)
act3 <- mx.symbol.Activation(fc3, name = "relu3", act_type = "relu")
drop3 <- mx.symbol.Dropout(act3, p = 0.2)

fc4 <- mx.symbol.FullyConnected(drop3, num_hidden = 1)

output <- mx.symbol.LinearRegressionOutput(fc4)

mx.set.seed(1)

logger <- mx.metric.logger$new()

model <- mx.model.FeedForward.create(output, X = train.array, y = yy,
ctx=mx.cpu(), num.round=100, array.batch.size=100, learning.rate=0.01,
momentum=0.9, wd=0.01, eval.metric=mx.metric.rmse,
epoch.end.callback = mx.callback.log.train.metric(5, logger))

```

23.6 Concluding Remarks

This chapter has focused on the method of maintenance effort estimation related to reliability analysis based on the deep learning by using fault big data on bug tracking system and GUI data. In particular, our method based on the deep learning has used the fault big data and GUI of OSS. Then,

the changing in GUI of OSS will depend on the development and management effort of OSS in the specified version. Considering the relationship between GUI and OSS development process, the UX/UI design of OSS will change with the procedure of OSS development.

Therefore, we have focused on the maintenance effort of OSS and GUI data. Moreover, we have shown the numerical illustrations of our method based on the deep learning.

Table 23.1 The actual data

Time	Effort	Time	Effort	Time	Effort
0	0	52	462245670	104	1067245901
1	11242	53	484592919	105	1069101192
2	81687	54	516998422	106	1069777567
3	151771	55	543744487	107	1070795028
4	419919	56	561214473	108	1072155427
5	988874	57	573350012	109	1073177882
6	1297656	58	583136767	110	1073861275
7	1942925	59	589657167	111	1075402199
8	2585842	60	600964595	112	1076259622
9	3095149	61	622307407	113	1077805430
10	3977554	62	640528209	114	1078665962
11	5204038	63	654526074	115	1079876837
12	6141696	64	669931665	116	1081619528
13	7737537	65	679991316	117	1083373725
14	10144016	66	690375293	118	1084607220
15	12307828	67	702043919	119	1086378005
16	14115921	68	714319511	120	1086910444
17	16013042	69	726665947	121	1087622126
18	20154925	70	739361520	122	1088157826
19	24526178	71	751194991	123	1088336474
20	34156018	72	761035714	124	1089232878
21	39611898	73	774980221	125	1091388752
22	43690308	74	789705980	126	1097870373
23	50142720	75	802263993	127	1105087734
24	55547182	76	816456806	128	1107983430
25	62986272	77	826571311	129	1109252640
26	71065755	78	837875789	130	1111799297
27	77509641	79	850103559	131	1112529511
28	87621023	80	864787201	132	1112895955
29	96656346	81	879264020	133	1113446237
30	106121972	82	896185115	134	1113997908
31	117197371	83	912643005	135	1114920303
32	131682720	84	925633810	136	1116215453
33	143577935	85	944251733	137	1118068558
34	160708649	86	960253359	138	1119183774
35	171961368	87	971028535	139	1120676344
36	188933756	88	984453051	140	1125734290
37	201072867	89	998855510	141	1133427805
38	210782860	90	1019053506	142	1135501260
39	223482044	91	1036872416	143	1138524374
40	233011838	92	1040159902	144	1139473017
41	244549868	93	1043484519	145	1140044481
42	256969749	94	1047156948	146	1140999352
43	268215630	95	1050376746	147	1141381659
44	278275066	96	1052963680	148	1142148952
45	292618694	97	1054753956	149	1142535167
46	311410264	98	1056896849	150	1142729155
47	331548254	99	1058717013	151	1143119002
48	351585734	100	1060874627	152	1144095627
49	381001674	101	1062541648	153	1146055373
50	412419834	102	1064553663	154	1147037419
51	440049846	103	1065225570	155	1147824308

(continued)

Table 23.1 (continued)

Time	Effort	Time	Effort	Time	Effort
156	1149401799	166	1160799310	176	1165826931
157	1149994837	167	1161205800	177	1166688184
158	1150984029	168	1161409329	178	1166904411
159	1151380520	169	1161818719	179	1167121734
160	1152772429	170	1162023803	180	1167343434
161	1152971629	171	1162642892	181	1167572822
162	1154367833	172	1163057500	182	1168047129
163	1155168131	173	1163264970	183	1168289321
164	1156371951	174	1163899094		
165	1159786977	175	1165397161		

Table 23.2 The results of goodness-of-fit comparison based on RMSE in terms of software effort

Compared models	RMSE
DFNN	4.1×10^{16}
DCNN	3.9×10^{16}

Thereby, we have found that our proposed method can estimate the development effort of OSS based on the GUI data and the fault big data on bug tracking system. In particular, this chapter shows the practical computational methods by using the MXNeT package the via Statistical Computing R.

References

1. Yamada, S.: Software Reliability Modeling: Fundamentals and Applications. Springer, Tokyo/Heidelberg (2014)
2. Lyu, M.R. (ed.): Handbook of Software Reliability Engineering. IEEE Computer Society Press, Los Alamitos, CA (1996)
3. Kapur, P.K., Pham, H., Gupta, A., Jha, P.C.: Software Reliability Assessment with OR Applications. Springer, London (2011)
4. Yamada, S., Tamura, Y.: OSS Reliability Measurement and Assessment. Springer International Publishing, Switzerland (2016)
5. Tamura, Y., Ashida, S., Matsumoto, M., Yamada, S.: Identification method of fault level based on deep learning for OSS. In: Software Engineering Research, Management and Applications, Studies in Computational Intelligence, vol. 654, pp. 65–76. Springer International Publishing, Switzerland (2016)
6. Tamura, Y., Yamada, S.: Comparison of big data analyses for reliable OSS. In: Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management, 2016, CD-ROM (Reliability and Maintenance Engineering 3) 5 pp.
7. Tamura, Y., Ashida, S., Yamada, S.: Fault identification tool based on deep learning for fault big data. In: Proceedings of the 3rd International Conference on Information Science and Security, pp. 69–72 (2016)
8. Zhou, Y., Davis, J.: OSS reliability model: an empirical approach. In: Proceedings of the Workshop on OSS Engineering (WOSSE), vol. 30(4), pp. 67–72 (2005)
9. Li, P., Shaw, M., Herbsleb, J., Ray, B., Santhanam, P.: Empirical evaluation of defect projection models for widely-deployed production software systems. In: Proceeding of the 12th International Symposium on the Foundations of Software Engineering (FSE-12), pp. 263–272 (2004)
10. Kingma, D.P., Rezende, D.J., Mohamed, S., Welling, M.: Semi-supervised learning with deep generative models. In: Proceedings of Neural Information Processing Systems (2014)

11. Blum, A., Lafferty, J., Rwebangira, M.R., Reddy, R.: Semi-supervised learning using randomized mincuts. In: Proceedings of the International Conference on Machine Learning (2004)
12. George, E.D., Dong, Y., Li, D., Alex, A.: Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Trans. Audio Speech Lang. Process.* **20**(1), 30–42 (2012)
13. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* **11**(2), 3371–3408 (2010)
14. Martinez, H.P., Bengio, Y., Yannakakis, G.N.: Learning deep physiological models of affect. *IEEE Comput. Intell. Mag.* **8**(2), 20–33 (2013)
15. Hutchinson, B., Deng, L., Yu, D.: Tensor deep stacking networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(8), 1944–1957 (2013)
16. The Apache Software Foundation, The Apache HTTP Server Project, <http://httpd.apache.org/>



Shigeru Yamada received his PhD from Hiroshima University, Japan, in 1985. From 1993/10 to 2018/3, he had been working at Tottori University in Japan. His research interests include software reliability engineering, quality/project management. He has authored 13 books on software reliability engineering and published over 600 reviewed technical papers.



Yoshinobu Tamura has been working as a Professor at the Graduate School of Sciences and Technology for Innovation, Yamaguchi University, Japan. His research interests include reliability assessment for open source software.



Kodai Sugisaki received his B.S.E. from Tokyo City University in 2019. His research interests included project management and open source software at Graduate School Integrative Science and Engineering, Systems Information Engineering, Tokyo City University.



Vulnerability Discovery Analysis in Software Reliability and Related Optimization Problems

24

P. K. Kapur and Saurabh Panwar

Contents

24.1	Introduction	457
24.1.1	Notations.....	459
24.2	Mathematical Modeling	459
24.2.1	NHPP-Based Vulnerability Discovery Models.....	459
24.2.2	Notations.....	460
24.2.3	User-Dependent VDM.....	460
24.2.4	Effort-Dependent VDM.....	461
24.2.5	Generalized Coverage-Dependent VDM.....	463
24.3	Empirical Analysis	465
24.3.1	Parameter Estimation and Goodness of Fit Criteria.....	465
24.3.2	Data.....	466
24.3.3	Validation Results of User-Dependent and Effort-Dependent VDMs.....	466
24.3.4	Validation Result of Generalized Coverage-Dependent VDM.....	468
24.4	Optimal Patch Time Release Policies	468
24.4.1	Assumption.....	468
24.4.2	Optimal Release and Patching Time of Software with Warranty.....	469
24.4.3	Optimal Patch Release Time for Both Faults and Vulnerabilities.....	473
24.4.4	Optimal Patch Release Time for Vulnerable Software Systems.....	476
24.5	Concluding Remarks	479
	References	479

Abstract

The recent rapid advancement in technology has affected the security of software products. The number of threats and cyber-attacks are intensifying both in number and in complexity. Therefore, software system requires protection against threats and vulnerabilities. When

defects in the software have an effect on the security of the software system, then these defects are called vulnerabilities. It is essential for vendors to rigorously identify and remove vulnerabilities present in the system. This chapter aims to explain the vulnerability discovery and patching process mathematically. Patch is a security update released by software developers to eliminate vulnerabilities from the system. Quantitative measures are discussed in the present study to predict the vulnerability discovery growth function by incorporating various attributes, namely, software users, operational effort, and coverage functions. Joint optimization problem for optimal software and patch time-to-market are also discussed with an aim of minimizing the cost functions. Numerical examples are provided to validate the mathematical models and minimization problem using the actual vulnerability data sets. The results indicate that the discussed models can objectively determine the vulnerability discovery paradigm. Moreover, the optimization models will assist the management team in optimal decision making pertaining to release time of software and security patch in the market.

Keywords

Software vulnerability · Prediction modeling · Patching · Joint optimization · Software time-to-market · Patch release time

P. K. Kapur (✉)
Amity Center for Interdisciplinary Research, Amity University, Noida, Uttar Pradesh, India

S. Panwar
Department of Operational Research, University of Delhi, Delhi, India

24.1 Introduction

In the last two decades, the software industry has witnessed tremendous advancements at an accelerated pace. The software is pervasive and has widespread utility. The human activities are notably dependent on the software. Indeed, the employment of the software has transformed humankind in every facet. Therefore, reliability and quality of software

system is most essential aspect of the software development life cycle. In software engineering literature, academicians and practitioners have suggested numerous software reliability growth models (SRGMs) to predict the reliability of software packages over their life cycle. SRGMs are mathematical models that assist testers in identifying the failure occurrence phenomenon and fault content of software systems at any point of time [1]. In software industry, the nonhomogenous process (NHPP)-based SRGMs are often applied to quantify the failure occurrences of software systems.

Software products are designed using varied procedures and are based on diverse technologies [2]. However, every piece of software has its own benefits as well as drawbacks. The main concern associated with the software industry is the security of the software packages. The prime characteristic of security is *confidentiality*, *integrity*, and *availability* [3]. Confidentiality refers to impediment of unauthorized disclosure of information. Integrity involves stopping of unauthorized alteration of information, and availability signifies the accessibility of critical information to the authorized unit. Absence of any of these three aspects from the software indicates security infractions in the system, which can cause loss and exploitation of crucial information. These security breaches in software are commonly known as software vulnerabilities.

In reference of software security, vulnerabilities are all about faults or oversights in the software system, which enables the hackers to sidestep the security measures and perform malicious activities to disrupt the confidentiality, integrity, and availability of the software program. Thus, vulnerabilities are a division of bugs present in the system, which affects the security of the software [4]. The menace caused by the security breaches has different severity degree, depending on the elements such as manipulation intricacy and attack-surface [5]. Several instances have been observed in the recent past in which software vulnerabilities have inflicted substantial harm to people and technological industries [6]. Presently, every software product in the market is confined to security infractions. According to Computer Emergency Response Team (CERT) [7], hundreds of new vulnerabilities are reported every week. National Vulnerability Database (NVD) [8] reports that overall 16,555 vulnerabilities are accounted for the year 2018, which is the largest figure until date. Therefore, discovering vulnerabilities is a preeminent activity that needs momentous human efforts.

Due to the continuous improvement in the information systems, software vulnerabilities pose a severe threat to the software engineers. Owing to the alarming intensity of these vulnerabilities, the research in the area of software security

has gained momentum from the past decade to provide security solutions to the industries. However, most of these studies are qualitative in nature. The quantitative aspect of vulnerability discovery process has not been given high importance in the software engineering literature. With the help of quantitative statistical analysis, team of software developers can predict the vulnerability discovery pattern more precisely by using the data-specific forecasting methods [9]. In recent past, few attempts have been carried out to characterize the security vulnerabilities mathematically. The first quantitative model for vulnerability discovery process was provided by Anderson [10], which is widely known as Anderson's thermodynamic model. Anderson [10] developed the thermodynamic vulnerability discovery model to measure the security in open and closed systems. The model illustrates the discovery rate based on mean time between failures (MTBF). According to their research, there is no difference between open and closed system in the end.

Rescorla [11] determined the economic effectiveness of finding and fixing the identified vulnerabilities for the company especially when they are discovered by the black hat. Their model is based on the nonhomogeneous Poisson process (NHPP) to assess the vulnerability discovery rate over time. Two statistical models, namely, Rescorla Exponential (RE) and Rescorla Linear or Quadratic (RL or RQ) model, were suggested by the author to fit the different growth function of vulnerability data. Further, to investigate the S-shaped behavior of vulnerability discovery process, Alhazmi and Malaiya [12] proposed the logistic vulnerability discovery model that measures the expected vulnerabilities discovered over time. They have also suggested the new metric known as vulnerability density to measure the security risk associated with the software. They segregated the vulnerability discovery process into three phases, namely, linear, learning, and saturation phase. Another significant contribution to the vulnerability discovery modeling literature was provided by Joh et al. [13]. They developed the vulnerability discovery model (VDM) using the Weibull distribution function. Their model is commonly known as Joh Weibull (JW) Model. The model depicts the asymmetric nature of vulnerability discovery rate because of the skewness present in probabilistic density function.

Recently, Kapur et al. [14] have provided a comparative study of vulnerability discovery modeling and its interdisciplinary nature. They have proposed two different VDMs and compared the prediction capability with existing models. Many other studies have also been proposed in the past literature to quantify the vulnerability discovery paradigm [6, 15–20]. The mathematical forms of some existing vulnerability discovery models important to the present study are given in Table 24.1.

Table 24.1 Existing vulnerability discovery models

Vulnerability discovery model	Vulnerability discovery rate	Mean value function
<i>Rescorla Exponential (RE) Model</i>	$\frac{d\Omega(t)}{dt} = A(B - \Omega)$	$\Omega(t) = B(1 - e^{-At})$
<i>Rescorla Linear/Quadratic (RL/RQ) Model</i>	$\frac{d\Omega(t)}{dt} = Bt + k$	$\Omega(t) = \frac{Bt^2}{2} + kt$
<i>Alhazmi Malaiya Logistic (AML) Model</i>	$\frac{d\Omega(t)}{dt} = A\Omega(B - \Omega)$	$\Omega(t) = \frac{B}{1 + BCe^{-AB}}$
<i>Joh Weibull (JW) Model</i>	$\frac{d\Omega(t)}{dt} = B \left\{ \frac{\alpha}{\beta} \cdot \left(\frac{t}{\beta}\right)^{\alpha-1} e^{-\left(\frac{t}{\beta}\right)^\alpha} \right\}$	$\Omega(t) = B \left\{ 1 - e^{-\left(\frac{t}{\beta}\right)^\alpha} \right\}$ or $\Omega(t) = B \{1 - e^{-At^\alpha}\}$

24.1.1 Notations

$\Omega(t)$	Expected number of vulnerabilities discovered in interval $[0, t]$
B	Initial number of vulnerabilities
A/γ	Vulnerability discovery rate
k, C	Integration constants
α	Shape parameter
β	Scale parameter

The defect identification process is affected by various testing and environment conditions. However, all the above-mentioned vulnerability discovery models are the function of time alone. Less or no attention is given to other attributes such as number of software end-users, intensity of operational efforts, and coverage function. Therefore, in this chapter, the influence of these critical factors on the vulnerability discovery process has been studied. The mathematical models describing the prediction of vulnerability discovery growth function on different factors have been described and further validated on the real-life vulnerability data.

Another important stream of literature in the field of software engineering has focused on the optimal release time of a software and patch time management. Developers must understand the market and release the software at an optimum time with minimum cost and maximum reliability. In software reliability literature, various studies have been designed to formulate optimal software time-to-market [21–27]. In addition, several researchers have also focused on patch time problem. Patch is a piece of code designed by the experts to update the software and remove the malicious faults and security breaches from the software system. However, the time to release the patch to its end-users requires crucial examination as it incurs huge cost in a form of testing, vulnerability discovery, goodwill loss, and market opportunity cost. Beattie et al. [28] formulated the cost minimization problem to determine the optimal time to release the patches. The cost function associated with their model considers the cost of attack due to the vulnerability and the cost of depraving due to flawed patches over time. Some authors have also distinguished the time of vulnerability disclosure (publication of vulnerability existence in the system) and patch release. In this regard, Telang and Wattal [29] explore the losses incurred to the software firms because of the public

disclosure of vulnerabilities. Arora et al. [30, 31] also suggested the optimal policy for software vulnerability disclosure and firm's patch release behavior. However, limitations of these studies are that they either have focused on software time-to-market policy or patch release time management. In this chapter, joint optimization of software time-to-market and patch release time policies is described. The objective of the present study is to predict the vulnerability discovery growth function with respect to factors such as number of end-users, effort consumption rate, coverage function, and time. In addition, the optimal policy for software and patch release time is discussed, which can facilitate the software vendors for optimal decision-making.

The rest of the chapter is organized in the following sections. Firstly, the mathematical modeling for vulnerability discovery process is discussed by incorporating different criteria such as software users, operational effort function, and coverage rate. The empirical analysis is described in the next section to assess the prediction capability and estimation efficiency of all the vulnerability discovery models. After that, the chapter deals with three different optimization problems with the aim of determining the optimal patch time-to-market. Numerical illustrations are also provided to comprehend the practical application of the release policies. Finally, the chapter is summarized using the concluding remarks.

24.2 Mathematical Modeling

24.2.1 NHPP-Based Vulnerability Discovery Models

In this chapter, the quantitative techniques described to assess the growth of software vulnerability discovery during the operational phase have been classified using nonhomogenous poison process (NHPP). Different attributes such as software users growth function, intensity of operational efforts, and coverage functions have been applied to illustrate the vulnerability discovery process of software products. According to the NHPP-based models, the software security failures occur at random times during operational phase caused by vulnerabilities lying hidden in the software, which arises due to the instructions executed by users. The counting process $\{N(t), t \geq 0\}$ of an NHPP is defined as:

$$\Pr [V(t) = n] = \frac{[\Omega(t)]^n}{n!} \cdot e^{-\Omega(t)}, \quad n = 0, 1, 2, \dots$$

where $\Omega(t)$ is the basic building block of all the NHPP models and it is defined as the expected number of vulnerabilities discovered by time t .

24.2.2 Notations

$\Omega(t)$	Expected number of vulnerabilities discovered by time t
\bar{V}	Fixed number of vulnerabilities present in the software.
\bar{S}_b	Potential users of a software product
$S_b(t)$	Cumulative number of software users by time t
$I_e(t)$	Expected number of actions performed or instructions executed
$W_e(t)$	Operational effort function; number of users who reports the presence of vulnerability in the system by time t
$C_f(W_e(t))$	Operational coverage function; the proportion of software covered for vulnerability discovery by time t

24.2.3 User-Dependent VDM

In this subsection, a user-dependent vulnerability discovery model is suggested [32]. The model is based on the key theory that total vulnerabilities identified in the software system are a function of the three critical factors: residual vulnerabilities present in the software system, number of instruction performed by the software users, and cumulative users of the software system. The model is described using the following assumptions:

- The software system fails randomly due to vulnerabilities present in the software.
- The total vulnerabilities identified at any instance are directly proportional to the undetected vulnerabilities in the software, number of instructions executed by the users, and the number of users of the system.
- The number of software buyers is a function of time that affects the vulnerability discovery process considerably.
- The vulnerability discovery process is modeled by the nonhomogeneous Poisson process (NHPP).
- Vulnerability discovery is a deterministic process in a continuous-state space.

Thus, the instantaneous vulnerability discovery at a time-point t can be described using the following differential equation [32]:

$$\frac{d\Omega(t)}{dt} = \left(\frac{d\Omega(t)}{dI_e(t)} \right) \cdot \left(\frac{dI_e(t)}{dS_b(t)} \right) \cdot \left(\frac{dS_b(t)}{dt} \right) \quad (24.1)$$

where time-dependent functions $\Omega(t)$, $I_e(t)$, and $S_b(t)$ clearly depend on the functions $I_e(t)$, $S_b(t)$, and t , respectively. The above-stated equation addresses that the vulnerability discovery in any software product is a function of number of instructions implemented, software users, and time. A detailed description of each component (or faction) of the model is presented below.

Component 1 The first component illustrates the rate of change in the number of vulnerabilities discovered with respect to the rate of change in number of instructions performed by the users. When software is released in the market, the developers continuously monitor and test its performance to identify any security breach present in the system. It has also been established that most of the vulnerabilities are interdependent [33]. Therefore, the probability of detecting other vulnerabilities increases when first vulnerability is identified. Thus, the total vulnerabilities can be classified into two classes: *unique* and *interdependent*.

Let *unique* vulnerabilities are discovered with rate x and y is the rate of discovery of *dependent* vulnerabilities, then the mathematical expression for the instantaneous vulnerabilities discovered at a time-point t is a function of two additive terms, that is:

$$\frac{d\Omega(t)}{dI_e(t)} = \left(x + y \cdot \frac{\Omega(t)}{\bar{V}} \right) \cdot (\bar{V} - \Omega(t)) \quad (24.2)$$

where $(\bar{V} - \Omega(t))$ are the vulnerabilities which are not identified as yet; the first term on the right hand describes the discovery of independent or unique vulnerabilities with rate x , and the second term represents the identification of dependent vulnerabilities with rate y , which is dependent on the proportion of already discovered vulnerabilities. It may be noted that the vulnerability discovery is an increasing function of the number of instructions executed. The number of vulnerabilities discovered increases as large number of instructions are executed.

Component 2 When users execute instructions on the software system, they may encounter some irregularities or security breaches. They report these discoveries to the developers for further investigation and timely removal of the vulnerabilities. Thus, the rate of change of instruction executed depends directly on the rate with which the number of users of the software system changes. Let ϕ be the proportionality constant, then the mathematical expression for the instantaneous change in the executed instructions at any time-point t becomes:

$$dI_e(t) = \phi dS_b(t) \quad (24.3)$$

$$\Rightarrow \frac{dI_e(t)}{dS_b(t)} = \phi \quad (24.4)$$

On substituting the expression of Eqs. (24.2) and (24.4) in Eq. (24.1), following mathematical expression is formed:

$$\frac{d\Omega(t)}{dt} = \left\{ \left(x + y \cdot \frac{\Omega(t)}{\bar{V}} \right) \cdot (\bar{V} - \Omega(t)) \right\} \cdot \{\phi\} \cdot \frac{dS_b(t)}{dt} \quad (24.5)$$

The above differential equation can be further expressed as:

$$\frac{\frac{d\Omega(t)}{dt}}{\frac{dS_b(t)}{dt}} = \left\{ \left(x + y \cdot \frac{\Omega(t)}{\bar{V}} \right) \cdot (\bar{V} - \Omega(t)) \right\} \cdot \{\phi\} \quad (24.6)$$

The above-equation can be solved using the boundary condition, $\Omega(t) = 0$, $S_b(t) = 0$ at $t = 0$, to attain the following closed-form solution:

$$\Omega(t) = \bar{V} \cdot \frac{(1 + h \cdot e^{-(x+y) \cdot S_b(t)})^\phi - (1 + h) \cdot e^{-(x+y) \cdot \phi \cdot S_b(t)}}{(1 + h \cdot e^{-(x+y) \cdot S_b(t)})^\phi} \quad (24.7)$$

(VDM1)

where $h = \frac{y}{x}$; $b = x + y$ and Eq. (24.7) represents the user-dependent vulnerability discovery model.

Modeling Software Buyers Growth Function

The third component of Eq. (24.1) relates to the change in the number of software users with time. Numerous expressions have been suggested to represent the relationship between the users and the time. One of the well-known expressions was given by Kenny [34], who has established that the growth rate of software users is a power function of the time. According, to his study, the usage of the software in the field environment is governed by a power law function, that is,

$$S_b(t) = \frac{t^{\lambda+1}}{\lambda + 1} \quad (24.8)$$

where λ is any real constant. However, the power function is seldom used in describing the relation between the users and the time. This is because the parameter of the expression is not interpretable in context of user's growth rate. Moreover, it does not differentiate the two classes of buyers, particularly, *innovators*, who buy the new product under the influence of mass-communication and *imitators*, who purchase a new

product under the influence of interpersonal communication. Therefore, in the present study, the mathematical expression given by Bass [35] is employed to differentiate the two groups of buyers categorically.

According to the Bass study, there exists a finite population of prospective users, say \bar{S}_b , who time increasingly purchases a product. Let p be the rate of innovation and q be the rate of imitation then the differential equation for instantaneous in the change the number of users at any time-point becomes:

$$\frac{dS_b(t)}{dt} = \left(p + q \cdot \frac{S_b(t)}{\bar{S}_b} \right) \cdot (\bar{S}_b - S_b(t)) \quad (24.9)$$

where $(\bar{S}_b - S_b(t))$ signify the probable users who are still left to purchase a software product and $S_b(t)$ is the cumulative software buyers by time t . It is believed that the vulnerabilities discovery increases with the increase in the number of instructions implemented at the user's end and on the other hand, number of executed instruction accelerated with the growth of software buyers. Thus, it can be interpreted that the growth function of software users affects the discovery of vulnerabilities. The above differential equation can be evaluated using the following integral structure:

$$\int dS_b(t) = \int \left(p + q \cdot \frac{S_b(t)}{\bar{S}_b} \right) \cdot (\bar{S}_b - S_b(t)) \cdot dt \quad (24.10)$$

On solving the Eq. (24.10) under the boundary condition, $S_b(t) = 0$ at $t = 0$, following expression for the cumulative software users by time t is formed:

$$S_b(t) = \bar{S}_b \cdot \left(\frac{1 - e^{-(p+q)t}}{1 + \frac{q}{p} e^{-(p+q)t}} \right) \quad (24.11)$$

where $\gamma = q/p$, $r = p + q$

Equation (24.11) denotes the mathematical expression for the total software users by time t . The user's growth can be used to evaluate the vulnerability discovery growth function.

24.2.4 Effort-Dependent VDM

This subsection explores the vulnerability discovery model by examining the consequences of operational effort and operational coverage function [36]. As the total efforts or resources consumed in the operational phase increase, more and more coverage can be provided to the software product. Thus, this increase in coverage will in turn accelerate the vulnerability discovery process. The model is based on the following key assumptions:

- The vulnerability discovery process is a function of number of vulnerabilities present in the system at that instance and the software coverage rate.
- The software coverage function directly depends on the efforts or resources (such as user/reporters) consumed in the operational phase.
- In addition, the rate of operational effort is a function of time.

Based on the above-mentioned assumptions, the instantaneous vulnerability discovery at a time-point t can be expressed as [36]:

$$\frac{d\Omega(t)}{dt} = \frac{d\Omega(t)}{dW_e(t)} \cdot \frac{dW_e(t)}{dt} \quad (24.12)$$

The differential equation described in Eq. (24.12) is a function of two components or fractions. The first component explains that the rate of vulnerability discovery at any time instant depends on the rate of change in operational efforts consumed at that point. The second component describes that the rate of change in efforts is a function of time. The mathematical expression for the two components is provided below:

Component 1 During operational phase, the amount of vulnerabilities discovered is directly proportional to the vulnerabilities present in the software at that instant. As the coverage of the software increased, more vulnerabilities are identified. Therefore, the change of vulnerability discovery with respect to change in operational efforts is considered dependent on the amount of software coverage, that is,

$$\frac{d\Omega(t)}{dW_e(t)} = \left(\frac{C'(W_e(t))}{z - C(W_e(t))} \right) \cdot (\bar{V} - \Omega(t)) \quad (24.13)$$

where $C'(W_e(t))$ is the coverage density function rate, $C(W_e(t))$ is coverage distribution function, z is the maximum proportion of software to be covered with $0 \leq z \leq 1$. Thus, $z - C(W_e(t))$ denotes the proportion of software, which remains uncovered. Coverage distribution functions can be modeled using the following functions:

- (a) When the coverage function is taken as *constant*, then

$$\left(\frac{C'(W_e(t))}{z - C(W_e(t))} \right) = b \quad (24.14)$$

On substituting, the value of Eq. (24.14) in Eq. (24.12), then the vulnerability discovery process can be described as:

$$\frac{d\Omega(t)}{dt} = b(\bar{V} - \Omega(t)) \frac{dW_e(t)}{dt} \quad (24.15)$$

Further, the differential equation (24.15) is solved using the boundary condition, $\Omega(t) = 0$, $W_e(t) = 0$ at $t = 0$ to get the following vulnerability discovery model:

$$\Omega(W_e(t)) = \bar{V}(1 - e^{-bW_e(t)}) \quad (\text{VDM2}) \quad (24.16)$$

Equation (24.16) describes the cumulative vulnerabilities discovered by time t when coverage function hazard-rate is constant.

- (b) When the coverage rate is assumed to follow *logistic* distribution function, then

$$\left(\frac{C'(W_e(t))}{z - C(W_e(t))} \right) = \frac{b}{(1 + h \cdot e^{-bW_e(t)})} \quad (24.17)$$

On substituting, the value of Eq. (24.17) in Eq. (24.12), then the vulnerability discovery process can be described as:

$$\frac{d\Omega(t)}{dt} = \frac{b}{(1 + h \cdot e^{-bW_e(t)})} \cdot (\bar{V} - \Omega(t)) \frac{dW_e(t)}{dt} \quad (24.18)$$

Further, the differential equation (24.18) is solved using the boundary condition, $\Omega(t) = 0$, $W_e(t) = 0$ at $t = 0$ to get the following vulnerability discovery model:

$$\Omega(W_e(t)) = \bar{V} \left(\frac{1 - e^{-bW_e(t)}}{1 + h e^{-bW_e(t)}} \right) \quad (\text{VDM3}) \quad (24.19)$$

Equation (24.19) describes the cumulative vulnerabilities discovered by time t when coverage function follows logistic distribution.

Modeling Operational Effort Function

The pace of almost all the software operations such as in-house testing, debugging faults, vulnerability discoveries is governed by the amount of resources utilized in the process. The second component of Eq. (24.12) represents the rate of change in operational efforts with respect to rate of change in time. In software reliability literature, effort function is modeled using the following distribution functions:

Exponential Effort When the effort consumption rate follows exponential distribution function, then operational effort function is defined as:

$$W_e(t) = \bar{W}(1 - e^{-\lambda t}) \quad (24.20)$$

where λ is a scale parameter specifying the rate with which efforts are being consumed and \bar{W} is the fixed amount of resources available for the vulnerability discovery process.

Weibull Effort When the effort consumption rate follows Weibull distribution function, then operational effort function is defined as:

$$W_e(t) = \bar{W} \left(1 - e^{-\lambda t^k}\right) \quad (24.21)$$

where λ is a scale parameter, k is a shape parameter, and \bar{W} is the total operational efforts to be utilized.

24.2.5 Generalized Coverage-Dependent VDM

In this subsection, the vulnerability discovery models discussed in earlier sections have been extended to include the effect of operational coverage, executed instructions, operational efforts, and time simultaneously. The model discussed in this section forms a generalized coverage-dependent model. Operational coverage is software metric that assists in evaluating the completeness and effectiveness of the usage process [37]. In general, coverage is a criterion to measure the thoroughness in executing the software system. It is considered that in the user environment, the probable vulnerable sites of the software product are covered using the efforts to discover vulnerabilities with some certainty. The model is based on the following assumptions:

- Vulnerabilities exist and uniformly spread in the software system over all potential vulnerable sites.
- When a potential vulnerable site is covered, using efforts at time t , the vulnerabilities at particular site are identified with certain probability.
- The expected vulnerabilities discovered in a small time-interval, say $(t, t + dt)$, are directly dependent on the number of vulnerabilities present in the system.
- The rate with which vulnerabilities are discovered is assumed a function of operational coverage.
- The operational coverage rate is a function of number of instructions executed.
- The number of executed instructions is considered effort dependent.
- The operational effort utilized in the vulnerability discovery process is a function of time.

In operational phase, the security breaches that are left unattended in the testing phase are discovered and patched. An operational coverage-based mathematical model is discussed to understand the underlying assumptions and predict the behavior of vulnerabilities. The model focuses on the effort spend in covering large proportion of the software for discovering vulnerabilities. Operational coverage is very important for both software developers and users. It can determine the effectiveness of a given operational effort. For developers, coverage information estimates the degree

of confidence of vulnerabilities that helps in vulnerability disclosure and patching. Coverage-based vulnerability discovery rate during the operational phase is defined as [38]:

$$\frac{d\Omega(t)}{dt} = \frac{d\Omega(t)}{dC_f(t)} \cdot \frac{dC_f(t)}{dI_e(t)} \cdot \frac{dI_e(t)}{dW_e(t)} \cdot \frac{dW_e(t)}{dt} \quad (24.22)$$

where the functions $\Omega(t)$, $C_f(t)$, $I_e(t)$, and $W_e(t)$ clearly depend on the functions $C_f(t)$, $I_e(t)$, $W_e(t)$, and t , respectively. A detailed description of each fraction on right-hand side is explained below:

Component 1 The instantaneous change of the vulnerability discovery rate with respect to operational coverage rate is directly dependent on the software vulnerabilities present in the software and indirectly proportional to uncovered proportion of software. A perfect coverage demonstrates that at infinity $C_f(\infty) = 1$, which is difficult to achieve in the operational phase. Consequently, for the modeling purpose, a target proportion of software, z has been taken that must be covered to identify vulnerabilities, where $0 < z < 1$. Moreover, the vulnerability discovery rate will enhance with increase in software coverage rate. Accordingly, the first component can be described as:

$$\frac{d\Omega(t)}{dC_f(t)} = A_1 \left(\frac{C'(W_e(t))}{z - C(W_e(t))} \right) (\bar{V} - \Omega(t)) \quad (24.23)$$

where A_1 is the proportionality constant representing the vulnerability discovery rate; $\left(\frac{C'(W_e(t))}{z - C(W_e(t))} \right)$ is the conditional probability that describes the rate with which the software is covered during the operational phase at any time t . Also, the operational coverage is a function of efforts employed in the operation phase.

Component 2 The instantaneous coverage rate corresponding to instantaneous number of executed instructions is taken as constant. As more and more instructions are performed on the software system, more software surface can be covered to discover vulnerabilities, that is,

$$\frac{dC_f(t)}{dI_e(t)} = \phi_1 \quad (24.24)$$

Component 3 The third component demonstrates the rate of change in executed instructions with respect to rate of change in operational efforts. Moreover, this change in execution rate of instructions per operational effort is considered constant. However, for some commercial software, the number of instruction executed is observed to be either increasing or decreasing, but it does not lead to much loss in generality. Thus, the third component is defined as:

$$\frac{dI_e(t)}{dW_e(t)} = \phi_2 \tag{24.25}$$

On combining the differentiable equations of above-mentioned components, and substituting Eqs. (24.23), (24.24), and (24.25) in Eq. (24.22), the following functional form is obtained:

$$\frac{d\Omega(t)}{dt} = \left\{ A_1 \left(\frac{C'(W_e(t))}{z - C(W_e(t))} \right) (\bar{V} - \Omega(t)) \right\} \cdot \{\phi_1\} \cdot \{\phi_2\} \cdot \frac{dW_e(t)}{dt} \tag{24.26}$$

The above differential equation can be further expressed as:

$$\frac{\frac{d\Omega(t)}{dt}}{\frac{dW_e(t)}{dt}} = A_1 \left(\frac{C'(W_e(t))}{z - C(W_e(t))} \right) (\bar{V} - \Omega(t)) \cdot \phi_1 \cdot \phi_2 \tag{24.27}$$

The above equation can be solved using the condition, $\Omega(0) = C(0) = 0$ to obtain the mean value function representing the expected number of discovered vulnerabilities by time t :

$$\Omega(W_e(t)) = \bar{V} \left(1 - \left(1 - \frac{C(W_e(t))}{p} \right)^{A_1 \phi_1 \phi_2} \right) \tag{24.28}$$

Equation (24.28) represents the total vulnerabilities discovered by time t when total $W_e(t)$ operational efforts have been expended.

Modeling Operational Effort Function

As mentioned before, the swiftness in the software operations is highly dependent on the amount of resources employed in the process. By utilizing efforts, the developers can meticulously identify security threats with an accelerated rate. Therefore, it is imperative to incorporate the effect of operational efforts for modeling the vulnerability discovery process. Thus, the fourth component in the differential equation (24.22) addresses the instantaneous rate of change in the operational efforts with respect to time. In present study, following distribution functions are applied to model the operational effort functions:

Weibull Effort When the effort consumption rate follows Weibull distribution function, then operational effort function is defined as:

$$W_e(t) = \bar{W} \left(1 - e^{-\lambda t^k} \right) \tag{24.29}$$

where λ is a scale parameter, k is a shape parameter, and \bar{W} is the total operational efforts to be utilized.

Logistic Effort When the effort consumption rate follows Logistic distribution function, then operational effort function is defined as:

$$W_e(t) = \bar{W} \left(\frac{1 - e^{-\lambda t}}{1 + \beta e^{-\lambda t}} \right) \tag{24.30}$$

where λ is a scale parameter and β is a learning parameter.

Modeling Operational Coverage Function

The operational coverage is assumed to follow following distribution functions:

- (a) When coverage function follows *Weibull distribution* function, then

$$C(W_e(t)) = p \left(1 - e^{-A_2 W_e(t)^h} \right) \tag{24.31}$$

where A_2 is a scale parameter and h is a shape parameter. Thus, the total vulnerability discovered (Eq. 24.28) becomes

$$\Omega(W_e(t)) = \bar{V} \left(1 - e^{-A_2 W_e(t)^{h A_1 \phi_1 \phi_2}} \right) \tag{VDM4} \tag{24.32}$$

Equation (24.32) describes the cumulative vulnerabilities discovered by time t when coverage function follows Weibull distribution function.

- (b) When coverage function follows *Logistic distribution*, then

$$C(W_e(t)) = p \left(\frac{1 - e^{-A_2 W_e(t)}}{1 + r e^{-A_2 W_e(t)}} \right) \tag{24.33}$$

where A_2 is a scale parameter and r is a learning parameter. Thus, the total vulnerability discovered (Eq. 24.28) becomes

$$\Omega(W_e(t)) = \bar{V} \left(1 - \left(1 - \left(\frac{1 - e^{-A_2 W_e(t)}}{1 + r e^{-A_2 W_e(t)}} \right)^{A_1 \phi_1 \phi_2} \right) \right) \tag{VDM5} \tag{24.34}$$

Equation (24.34) describes the cumulative vulnerabilities discovered by time t when coverage function follows Logistic distribution function.

24.3 Empirical Analysis

Empirical analysis is the most crucial phase in the model development process. Use of a better-fit software reliability growth model provides appropriate solutions to the fundamental engineering and technical problems. This section focuses on the validation of the above-discussed vulnerability discovery models and comparison of their prediction efficiency with existing models.

24.3.1 Parameter Estimation and Goodness of Fit Criteria

The nonlinear least square estimation procedure is applied for parameter estimation purpose. Least square estimation is an effective measure for small and medium size samples. For model validation, the unknown parameters, that is, the number of vulnerabilities that are potentially discovered, vulnerability discovery rate, number of software users buying the product, and software usage rate is estimated using the software known as Statistical Package of Social Sciences (SPSS) tool. If the expected value of the total vulnerability discovered by time t is given by $\Omega(t)$, then the least square estimators of the parameters of the model can be defined as:

$$Z = \sum_{i=1}^n \left[\widehat{\Omega}(t_i) - \Omega(t_i) \right]^2 \quad (24.35)$$

where n is the number of observations; $\widehat{\Omega}(t)$ is the predicted value, and $\Omega(t)$ is the actual value of the expected number of vulnerabilities discovered in the software.

The prediction ability of the model is said to be accurate if its simulating behavior matches with the actual software data. Therefore, to compare the goodness-of-fit for different vulnerability discovery models, five statistical measures such as Bias, Mean Square Error (MSE), Mean Absolute Error (MAE), Root Mean Square Prediction Error (RMSPE), and Coefficient of Determination (R^2) are applied.

- (a) **Bias**: It is the sum of the difference between the predicted and actual value. Lower the value of bias, better is the model fit to the data. If the predicted and actual value is same, then the model is known as unbiased model.

$$\text{Bias} = \frac{\left(\sum_{i=1}^n \left[\widehat{\Omega}(t_i) - \Omega(t_i) \right] \right)}{n} \quad (24.36)$$

where n is the number of observations

- (b) **Mean Square Error (MSE)**: It refers to the average value of the deviation between the estimated value and the actual value as follows:

$$\text{MSE} = \frac{\left(\sum_{i=1}^n \left[\widehat{\Omega}(t_i) - \Omega(t_i) \right]^2 \right)}{n} \quad (24.37)$$

where n is the number of observations. The lower MSE represents less fitting error, thus superior goodness of fit.

- (c) **Mean Absolute Error (MAE)**: It measures the closeness of prediction and the eventual outcomes. It is calculated as the absolute difference between the actual and predicted values divided by the difference of number of observations and degree of freedom, that is,

$$\text{MAE} = \frac{\left(\sum_{i=1}^n \left| \widehat{\Omega}(t_i) - \Omega(t_i) \right| \right)}{n} \quad (24.38)$$

- (d) **Root Mean Square Prediction Error (RMSPE)**: It calculates the closeness with which a model predicts the actual observations.

$$\text{RMSPE} = \sqrt{(\text{Bias}^2 + \text{Variation}^2)} \quad (24.39)$$

Lower value of RMSPE Error signifies high predictive accuracy of the model.

- (e) **Coefficient of Determination (R^2)**: It evaluates the percentage of the total variation about the mean accounted for the fitted curve. It ranges in value from 0 to 1. It is calculated as:

$$R^2 = 1 - \frac{\text{residual SS}}{\text{corrected SS}} \quad (24.40)$$

Small values indicate that the model does not fit the data well. The larger the value, the better the model explains the variation in the data.

- (f) **Predictive Ratio Risk (PRR)**: It measures the error between actual and estimated values using the following formula:

$$\text{PRR} = \sum_{i=1}^n \left\{ \frac{\left(\Omega(t_i) - \widehat{\Omega}(t_i) \right)^2}{\Omega(t_i)} \right\} \quad (24.41)$$

- (g) **Akaike Information Criterion (AIC)**: This criterion judge the quality of a set of statistical models to each other. It is evaluated using the following formula:

$$AIC_c = n * \ln(\text{likelihood}) + 2 * K + ((2 * K * (K + 1)) / (n - K - 1)) \quad (24.42)$$

where \ln is the natural algorithm; K is the number of parameters in the model and n is the number of observations.

- (h) **Bayesian Information Criterion (BIC):** It is relatively similar to the AIC. Nevertheless, it describes the over fitting issue by adding a penalty term for the number of parameters in the model. It is defined as:

$$BIC = n * \ln(\text{likelihood}) + K * \ln(n) \quad (24.43)$$

24.3.2 Data

The validation of a model requires some historical data of actual fault count of commercial software project, some metrics, and statistical tools. The first requirement, that is, the historical data, has been obtained from the governmental and private repositories of software vulnerabilities. The data sets have been obtained from the Common Vulnerability Exposure repository [39]. The vulnerability data set of several commercial software applications that are being used very often by software users has been chosen. The detailed description of data sets is described below:

1. **Oracle VM VirtualBox (DS 1):** It is a leading corporation, which is completely integrated with cloud applications and platform services. VirtualBox is free and open-source software (FOSS) whose source code is available for public access. Open-source software packages are type of computer software, which can be modified and shared by its users [40]. In this data set, 66 vulnerabilities were discovered from Jan 2011 to Jan 2017 by 29 software users or buyers. These vulnerabilities belong to Oracle VM Virtual box version 4.x and 5.x. Oracle Virtual box version 4.0 was initially released on 22 December 2010 and its version 5 was released on 9 July 2015.
2. **Google Chrome (DS 2):** The Google chrome is a web browser that offers web services to its clients by connecting them to the Internet. Due to immense dependency on web platforms for day-to-day activities, these platforms are broadly used and as a result, the probability of web attack on Google chrome is quite high. Google Chrome has more than 1300 vulnerabilities in total that are published in CVE until May 2016. For the present analysis purpose, data set has been filtered by “DOS Overflow” vulnerability type with the severity greater than 8 and overall 47 vulnerabilities were observed that were reported by 46 users between September 2008 and May 2016. The objective is to concentrate on only those vulnerabilities that have the potential to harm the developer

economically; therefore, the vulnerabilities with severity level greater than 8 are considered.

24.3.3 Validation Results of User-Dependent and Effort-Dependent VDMs

Firstly, the parameters of the user growth function given in Eq. (24.11) are estimated using the data of actual users of the software. The result is listed in Table 24.2. Moreover, the results of goodness-of-fit criteria such as Mean Square Error (MSE), Coefficient of Determination, R^2 are also evaluated and the result is summarized in Table 24.2. Further, Fig. 24.1a provides the goodness-of-curve for the predicted software users and actual buyers. Secondly, the parameters of the operational effort function given in Eqs. (24.20) and (24.21) are estimated using the actual effort data for the two dataset. Data consist of actual number of reporters over time. Corresponding parameter estimation values are provided in Table 24.2. The results provided in Table 24.2 show that all the estimated coefficients are highly reliable and significant. Moreover, the values of statistical performance for all vulnerability discovery models as provided in Table 24.2 establish the predictive power and prediction efficiency of the vulnerability discovery models. Graphically, the goodness-of-curves for two efforts functions with actual effort data are represented in Fig. 24.1b.

Further, the model parameters are estimated using the vulnerability data. Moreover, the model fitness with the actual data is evaluated and the prediction efficiency of the present model is further compared with the previously established vulnerability models such as Anderson Thermodynamic Vulnerability Discovery model, Rescorla exponential model, Rescorla quadratic model, Alhazmi Malaiya Logistic model, and Joh model. The parameter values obtained during the fit and the corresponding goodness-of-fit criteria for the data set DS1 are given in Table 24.3. The results of the statistical measures as displayed in Table 24.3 depict that all the suggested vulnerability discovery models have high prediction efficacy. Goodness-of-fit curves for VDM1, VDM2, and VDM3 are also illustrated graphically in Fig. 24.2.

Table 24.2 Parameter estimation result for software users and operational effort functions

$S_b(t) \sim \text{Logistic Growth function}$					
VDM1	\bar{S}	γ	r	MSE	R^2
	43.167	26.004	0.049	0.688	0.989
$W_e(t) \sim \text{Exponential Effort}$					
VDM2	\bar{W}	λ	K	MSE	R^2
	624.093	4.4e(-4)	–	8.124	0.873
$W_e(t) \sim \text{Weibull Effort}$					
VDM2	\bar{W}	λ	k	MSE	R^2
	99.114	6.1e(-5)	1.969	0.825	0.987

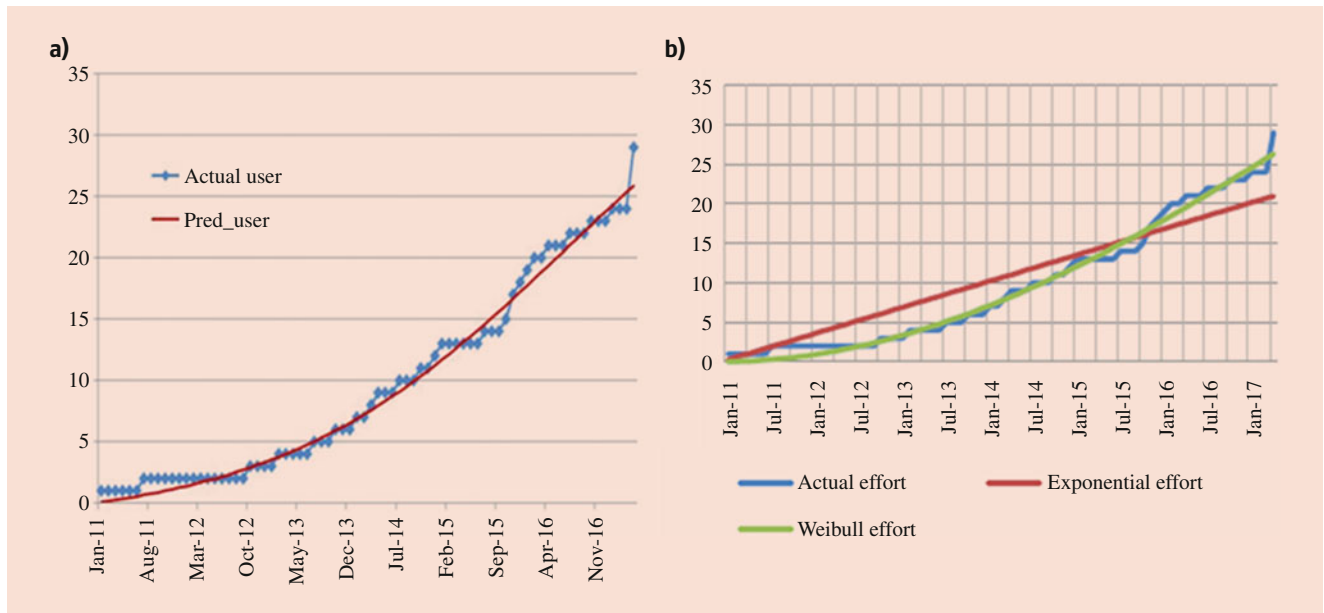


Fig. 24.1 Fitting Oracle VM virtual box (a) software buyers to the software user’s growth model and (b) effort function to the operational effort growth model

Table 24.3 Estimation results for VDM1 and VDM2 on Oracle VM Virtual Box (DS1) data set

Model	MSE	R ²	Bias	MAE	RMPSE	Parameter estimates
Rescorla Exponential (RE) model	65.008	0.82	0.002	0.0026	0.023	$B = 14720 A = 0.00004$
Rescorla Quadratic (RQ) model	5.253	0.985	-0.01	0.0148	0.129	$B = 0.023 k = -0.68$
Alhazmi Malaiya Logistic (AML) model	4.185	0.989	0.001	0.0013	0.011	$B = 71.653 A = 0.001, C = 1.224$
Joh Weibull (JW) Model	4.592	0.987	-9.1	0.1740	0.1106	$B = 92.676 A = 8.22 e(-6), \alpha = 2.711$
VDM1	3.898	0.99	-0.002	0.0031	0.0265	$\bar{V} = 70.22, b = 0.132 h = 4.56, \phi = 1.002$
VDM2 (Exponential Operational Effort)	65.143	0.819	0.0183	0.019	0.1642	$\bar{V} = 15791.7 b = 1.37e(-4)$
VDM2 (Weibull Operational Effort)	8.364	0.977	-0.009	-0.009	0.0867	$\bar{V} = 6934.859 b = 3.2e(-4)$
VDM3 (Exponential Operational Effort)	4.337	0.988	-0.012	0.0128	0.1114	$\bar{V} = 75.010 b = 0.263, h = 68.08$
VDM3 (Weibull Testing Effort)	4.575	0.987	-0.144	0.15	0.1719	$\bar{V} = 75.433 b = 0.124, h = 6.167$

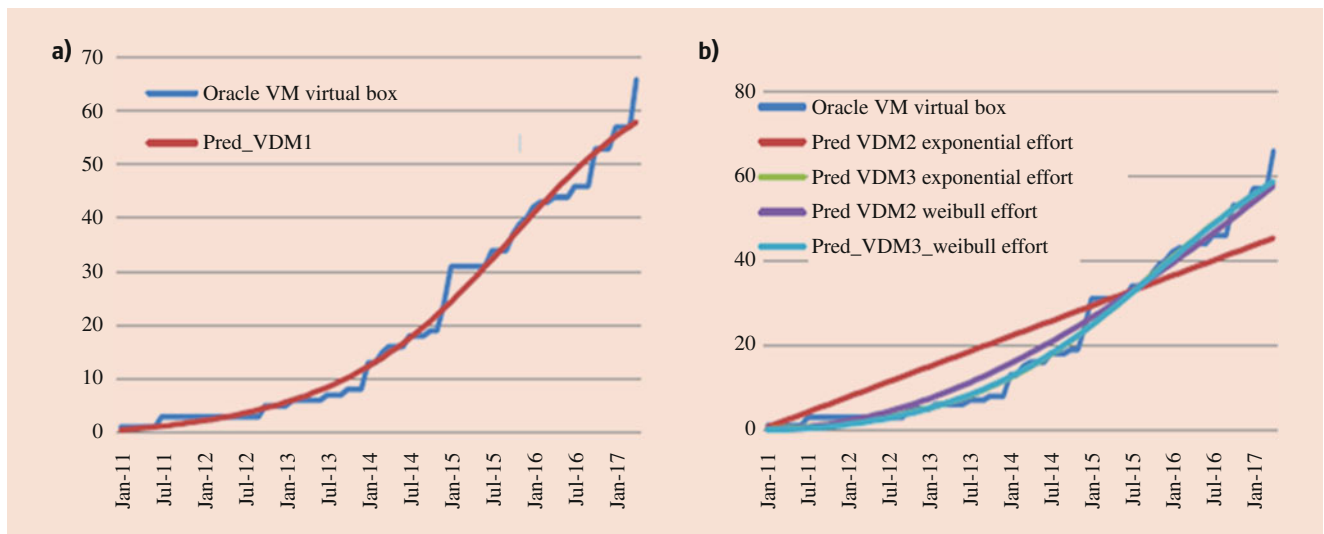


Fig. 24.2 Goodness of fit for Oracle VM virtual box (a) VDM1 (b) VDM2 & VDM3

The Rescorla exponential (RE) model for vulnerability discovery process is equivalent to the Goel-Okumuto (GO) model for software reliability growth function. In this model, the number of vulnerabilities discovered per unit time assumes to follow a Poisson process. From Table 24.3, it can be interpreted that the model does not provide good fit to the actual vulnerability data of Oracle virtual Machine virtual box with a R^2 value of 0.82. Rescorla Quadratic (RQ) model is also known as Rescorla linear model as it follows a uniform trend. The model provides reasonably good fit to the Oracle virtual box data set with the R^2 value of 0.985. Another established VDM is Alhazmi Malaiya Logistic (AML) model that examines the change in the cumulative number of vulnerabilities by the number of remaining vulnerabilities and the share of the installed base. The AML model follows an S-shaped pattern that contains three phases: learning phase, linear phase, and saturation phase. From the goodness-of-fit measures results, it can be deduced that the model provides good fit to the data with R^2 value of 0.989. The Joh vulnerability discovery model considers the Weibull distribution to model an asymmetric probability density function. This model is also supposed to follow the S-shaped pattern like AML model. It provides good fit to data with R^2 value of 0.987. It has been observed that the estimated and the actual data are following the same trend that shows a significant fit. The results in the Table 24.3 also suggest that for Oracle VM Virtual box, the model with least MSE and highest R^2 value is user-dependent vulnerability discovery model (VDM1), Joh model has least bias value, VDM2 with Weibull effort has the least MAE, while the least value of RMSPE is obtained by AML model.

24.3.4 Validation Result of Generalized Coverage-Dependent VDM

Initially, the parameters of the operational effort function given in Eq. (24.29) and (24.30) are estimated and the results are provided in Table 24.4. Additionally, Table 24.5 lists the estimated values of VDM3 and existing vulnerability discovery models such as RE, RL, and AML. Also, the goodness-of-fit measures such as R^2 , MSE, Bias, AIC, and PRR are calculated for the all the models. Results are summarized in Table 24.5. From the results displayed in Table 24.5, it can be interpreted that the value of MSE for all four VDMs provides good fit to data with low values of MSE, PRR, AIC, and BIC and significantly high value of R^2 . Further, Fig. 24.3 graphically depicts the fitting capability of different VDMs to the Google Chrome data set.

Table 24.4 Estimated values of operational effort function

Effort function	\bar{W}	λ	k/β	MSE	R^2
Weibull	40.672	0.001	2.116	2.414	0.989
Logistic	40.686	0.102	16.84	2.55	0.988

24.4 Optimal Patch Time Release Policies

In operational phase, vulnerability discovery by the developers is an ongoing process and does not end until the software is completely out of use or taken over by a new software. Thus, software is continuously monitored for new threats and flaws. New threats keep popping up and resources are required to correct and completely remove them. As a result, some coding is done to eliminate the threats in the form of a patch. The patch is then applied to the software and after testing, the patch is commercially released for the users. Patches are often released to secure software from new threats and identified flaws.

To evaluate the optimal time to release a patch keeping the cost expenditure minimum poses an insightful study for the software companies. Consequently, in the present section optimal patch release policies are discussed that may provide managerial aid and actionable recommendations to the software firms. There has been substantial argument as to when a company should disclose the discovered vulnerabilities and accordingly release a patch for the users. The software patch release time is a critical issue for the policy-makers that depends on various decisive factors such as companies' desired risk level, vulnerabilities discoverers, competence of patch testing, and market opportunity loss due to delay in software patch. The optimization problems discussed in this section yield optimal patch release-time situation. The optimization models are formulated with an aim of keeping the development cost and risk level minimum and high product reliability. The chapter covers the optimization problem of patch release time for faults and vulnerabilities.

24.4.1 Assumption

The optimization models are based on the following key assumptions:

- The fault and vulnerability discovery process follows a nonhomogeneous Poisson process (NHPP) phenomenon.
- The time lag between the fault detection and its removal is negligible.
- Faults are identified according to a Poisson process with mean value function (MVF) $m(t)$.

Table 24.5 Estimates values of different vulnerability discovery models for Google Chrome data set

Model	MSE	PRR	R^2	AIC_C	BIC	Parameter estimates
Rescorla Exponential (RE) model	21.372	11.616	0.989	281.624	289.128	$B = 55.797, A = 0.019$
Rescorla Quadratic (RQ) model	85.704	53.325	0.953	405.981	411.001	$B = 0.11, k = 0.982$
Alhazmi Malaiya Logistic(AML) model	21.372	11.616	0.989	281.624	289.128	$B = 41.333, A = 0.003, C = 0.528$
VDM4 (Weibull Testing Effort)	2.667	174.417	0.988	96.068	110.128	$\bar{V} = 169.219, h = 0.975$ $A_1 = 0.028, A_2 = 0.281$ $\phi_1 = 0.939, \phi_2 = 1.029$
VDM4 (Logistic Testing Effort)	2.54	28.0740	0.988	91.609	105.668	$\bar{V} = 122.042, h = 1.018$ $A_1 = 0.272, A_2 = 0.027$ $\phi_1 = 1.108, \phi_2 = 1.192$
VDM5 (Weibull Testing Effort)	2.661	216.149	0.988	95.86	109.92	$\bar{V} = 108.615, r = 0.167$ $A_1 = 0.156, A_2 = 0.128$ $\phi_1 = 0.781, \phi_2 = 0.781$
VDM5 (Logistic Testing Effort)	2.526	29.7027	0.989	91.121	105.181	$\bar{V} = 110.584, r = 0.204$ $A_1 = 0.022, A_2 = 0.21$ $\phi_1 = 1.12, \phi_2 = 2.241$

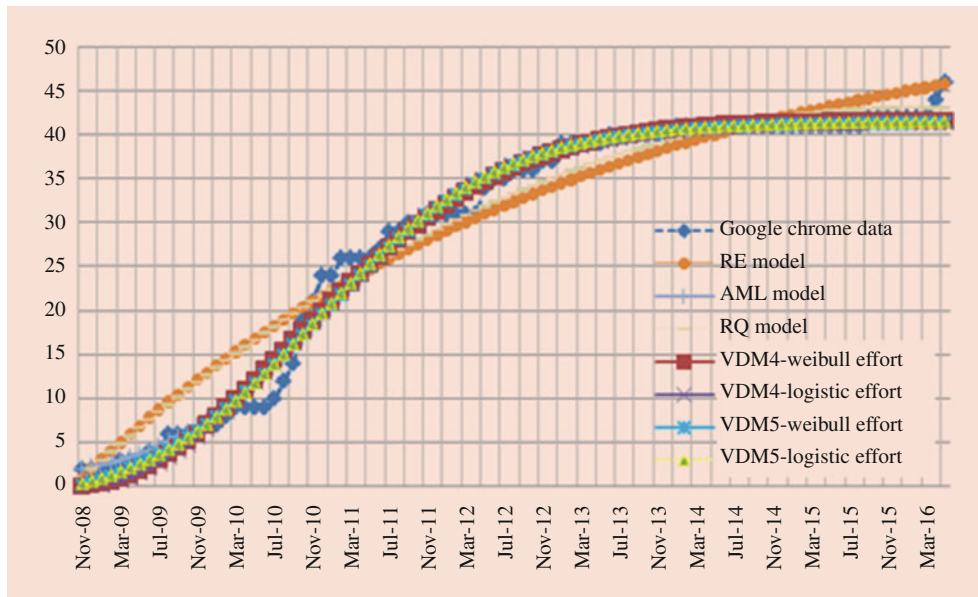


Fig. 24.3 Fitness of VDMs with Google Chrome Data

- Vulnerabilities are also discovered with respect to Poisson process with mean value function $\Omega(t)$
- The debugging process of defects is considered perfect, that is, faults and/or vulnerabilities are removed perfectly without introducing new faults and/or vulnerabilities in the system.
- There is fixed and finite number of defects in the system.
- Every software product has finite life cycle.
- Cost of patch development and delivery to the end-users is negligible.
- Market opportunity cost is assumed monotonically increasing and twice continuously differentiable convex function of the software release time.

- In postpatch release phase, no further patch will be released in the market for remaining defects rather a new upgraded version of the software will be released later by the company by removing the discovered vulnerabilities.

24.4.2 Optimal Release and Patching Time of Software with Warranty

This subsection presents a study to determine the optimal time to release a software and security patch when product is under warranty period [41]. The total life span of software is considered finite and it lies between the interval $[0, T_{lc}]$. The

entire software lifecycle is broadly divided into two phases, namely, testing phase $[0, \tau]$ and operational or maintenance phase $[\tau, T_{lc}]$. A warranty is offered on the software product at the time of the purchase so that if any failure occurs within the warranty period, it is rectified by the software vendors without any charge to the users. In operational phase, software is continuously tested for any security threats. Also, the manager's idea is to release a patch in the warranty period itself for the vulnerabilities discovered by that time. Thus, maintenance phase is further divided into prepatch release phase under warranty $[\tau, \tau_1]$, postpatch release phase under warranty $[\tau_1, \tau + w]$, and operational phase after warranty $[\tau + w, T_{lc}]$. The goal of the current optimization problem is to minimize the overall cost associated with the fault and vulnerability discovery and testing process with a constraint to maintain a desired reliability level.

Mathematical Modeling

In this subsection, the mathematical structure of the expected number of faults encountered in the entire software life cycle is studied in different phases. The pictorial representation of phase-wise description of software life cycle is given in Fig. 24.4.

Phase 1: Pre-release Testing Phase $[0, \tau]$

During this period, software is rigorously tested by the developers for any bugs present in the system. This is the most critical phase of software development life cycle because the stability, availability, and reliability of the software are determined based on testing process. If any test case is failed to perform during this phase, then the probability of failures increases in the operational phase. Therefore, vital importance is given to this phase by the software developers. The expected number of faults identified during this phase can be described using following expression:

$$m(\tau) = aF_1(\tau) \tag{24.44}$$

where a is the total fault content present in the software; $F_1(\tau)$ describes the fault distribution function of the testing team; τ is the release time of software in the market. It is further assumed that the faults detection phenomenon follows an exponential distribution pattern, that is,

$$m(\tau) = a(1 - e^{-b\tau}) \tag{24.45}$$

where b is the fault detection rate and a is the potential fault content present in the system.

Phase 2: Prepatch Release Phase Under Warranty $[\tau, \tau_1]$

After the software is released in the market at time τ , users meticulously identify flaws remaining in the system during its execution and report it to the developers. In this phase, a warranty is provided to the users to assure that if any discrepancy is observed then it will be solved by the developers without any charge. The faults detected during this phase are rectified by the developing team and a patch in the form of piece of code is developed to eliminate the faults. Thus, expected faults identified during this phase become:

$$m(\tau_1 - \tau) = a(1 - F_1(\tau))F_2(\tau_1 - \tau) \tag{24.46}$$

where $a(1 - F_1(\tau))$ are the leftover bugs from the phase 1 that are left unidentified in the pre-release testing phase and $F_2(\tau_1 - \tau)$ is the distribution function of fault detection by the users. Now, when fault detection phenomenon follows exponential distribution function, then mean value function of faults detection is expressed as:

$$m(\tau_1 - \tau) = ae^{-b\tau} (1 - e^{-b \cdot r_1 \cdot (\tau_1 - \tau)}) \tag{24.47}$$

where $b \cdot r_1$ ($0 < r_1 \leq 1$) is the combined fault detection rate of all the users. Since the competency of users in detecting the faults is less as compared to the testing team, the fault detection rate of users is considered some proportion of tester's bug detection rate.

Phase 3: Postpatch Release Phase Under Warranty $[\tau_1, \tau + w]$

For strengthening the position in the market, Software Company makes sure to provide a patch to its users for updating the system within the warranty period. It is considered that τ_1 is the optimal time when company should release its patch. Now, on receiving the patch, users successfully install it to update their product. However, not all faults can be fixed by a single patch. Therefore, there may be few faults that remained undetected in the previous phases and may be identified by the users after time-point τ_1 . It is assumed that the fault detection rate of users improves during postpatch release phase as they have attained experience through previous

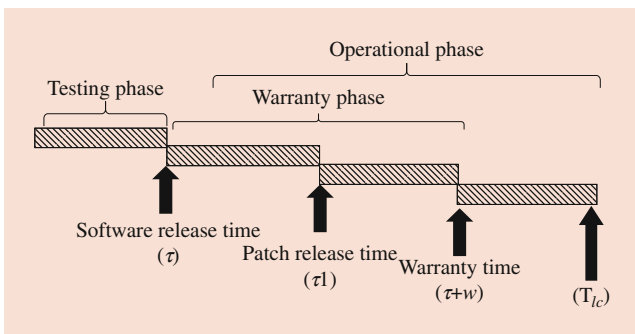


Fig. 24.4 Phase-wise description of software life cycle for fault discovery

phase $[\tau, \tau_1]$. Thus, the bug detection effectiveness increases during this phase. The expected number of bugs identified in third phase may be expressed as:

$$m((\tau + w) - \tau_1) = a(1 - F_1(\tau))(1 - F_2(\tau_1 - \tau)) F_3((\tau + w) - \tau_1) \quad (24.48)$$

where $a(1 - F_1(\tau))(1 - F_2(\tau_1 - \tau))$ represents the proportion of bug content which left undetected in the previous phase. Moreover, it is assumed that the bug detection phenomenon follows exponential distribution function. Then, the expected number of bugs detected during this phase becomes:

$$m((\tau + w) - \tau_1) = ae^{-b\tau} e^{-b \cdot r_1 \cdot (\tau_1 - \tau)} (1 - e^{-br_2(\tau + w - \tau_1)}) \quad (24.49)$$

where $b \cdot r_2$ is the increased bug detection rate of the users in phase 3.

Phase 4: Postwarranty Phase $[\tau + w, T_{lc}]$

It is assumed that warranty is terminated at a fixed time-point $(\tau + w)$. Generally, companies abort their free support to debug the reported faults after the warranty period is over. However, it has been observed that sometimes firms sign an agreement with the software users to correct software failures due to the faults present in the system throughout the product life cycle. Thus, the users continuously monitor and report the bugs to the developers for removal. The mean value function of faults detection during this phase can be modeled as:

$$m(T_{lc} - (\tau + w)) = a(1 - F_1(\tau))(1 - F_2(\tau_1 - \tau)) (1 - F_3((\tau + w) - \tau_1)) F_4(T_{lc} - (\tau + w)) \quad (24.50)$$

where $a(1 - F_1(\tau))(1 - F_2(\tau_1 - \tau))(1 - F_3((\tau + w) - \tau_1))$ correspond to the proportion of remaining faults and $F_4(T_{lc} - (\tau + w))$ is the cumulative fault detection function of the user in this phase. Now, the mean value function when fault detection process follows exponential distribution can be re-written as:

$$m(T_{lc} - (\tau + w)) = ae^{-b\tau} e^{-b \cdot r_1 \cdot (\tau_1 - \tau)} e^{-b \cdot r_2 \cdot (\tau + w - \tau_1)} (1 - e^{-b \cdot r_3 \cdot (T_{lc} - (\tau + w))}) \quad (24.51)$$

Cost Modeling

The following cost components are considered for modeling the cost function:

(a) **Testing cost** : It is the cost to execute testing of each software unit. It includes all the effort expends in developing and running the test cases. In software literature, this cost is assumed to increase linearly with time. Thus, if C_1 is the per unit testing cost, then total testing cost can be defined as:

$$C_{\text{testing_cost}}(t) = C_1 \tau \quad (24.52)$$

(b) **Market Opportunity Cost**: Before initializing the software development project, every company performs a market survey to investigate the expected profit and revenue gained by introducing a new software product. If a firm fall short in delivering the software at the scheduled launch time then the market opportunities are missed. Therefore, a cost in a form of market opportunity is implemented in the cost model, which is a nonlinear increasing function of release time, that is,

$$C_{\text{market_opportunity}}(t) = C_2 \tau^2 \quad (24.53)$$

where C_2 is the market opportunity cost per unit time.

(c) **Cost of fault debugging during testing phase**: It is the cost incurred in fixing the detected faults during the testing period. It is assumed that the debugging cost is directly proportional to the number of faults identified by the testers. If C_3 is the per unit debugging cost during testing phase then the overall cost of debugging in testing phase is represented as:

$$C_{[0, \tau]}(t) = C_3 m(\tau) \quad (24.54)$$

(d) **Cost of fault debugging in prepatch release phase**: After the release of the software in the market, it is in the operational environment where the faults are continuously discovered and reported by the users. The potential loss possessed by developers during this phase is huge as it includes the liability cost, loss of goodwill due to customer's dissatisfaction, the cost of developing and distributing patches, and the cost of revenue loss when failures are observed within warranty, etc. If C_4 is the per unit fault debugging cost in this phase, the total cost incurred by the testing team in prepatch release patch becomes:

$$C_{[\tau, \tau_1]}(t) = c_4 \cdot m(\tau_1 - \tau) \quad (24.55)$$

(e) **Cost of fault debugging in postpatch release phase**: Even after the release of first patch, some proportion of total faults remains undetected in the software. As software is still in the warranty period, developers have to

correct the failure causing bugs. If C_5 is the per unit bug removal cost in this phase, then effective cost incurred in postpatch release warranty phase is given as:

$$C_{[\tau_1, \tau+w]} = C_5 m (\tau + w - \tau_1) \tag{24.56}$$

(f) **Cost of fault debugging in postwarranty phase:** Even after the termination of warranty at time-point $(\tau + w)$, software developers keep on debugging the faults reported by the users. Moreover, the postwarranty period is much longer than the other software lifecycle phases, which results in high utilization of resources during this period. Therefore, per unit cost of bug correction is highest during this phase. The total debugging cost in postwarranty phase when C_6 cost of fault removal per unit can be defined as:

$$C_{[\tau+w, T_{lc}]} = C_6 \cdot m (T_{lc} - (\tau + w)) \tag{24.57}$$

Thus, the total cost can be obtained by adding all the cost components, that is,

$$\begin{aligned} \text{Total_Cost, } C(t) = & C_1 \tau + C_2 \tau^2 + C_{[0, \tau]}(t) + C_{[\tau, \tau_1]}(t) \\ & + C_{[\tau_1, \tau+w]}(t) + C_{[\tau+w, T_{lc}]}(t) \end{aligned} \tag{24.58}$$

Based on the aforementioned assumptions and modeling framework, the following constrained optimization problem is defined with an objective of minimizing the total cost and maintaining the certain level of software reliability.

$$\begin{aligned} \text{Minimize } C(t) = & C_1 \tau + C_2 \tau^2 + C_{[0, \tau]}(t) + C_{[\tau, \tau_1]}(t) \\ & + C_{[\tau_1, \tau+w]}(t) + C_{[\tau+w, T_{lc}]}(t) \end{aligned} \tag{24.59}$$

$$\text{Subject to, } R(x|t) \geq R_0 \tag{24.60}$$

where $R(x|t)$ is the conditional reliability function that states software will perform failure-free in a small time interval $[t, t + x]$ under specified user environment and mathematically it is defined as:

$$R(x|t) = e^{-\{m(t+x)-m(t)\}} \tag{24.61}$$

Numerical Example

The constrained optimization problem defined in the previous section has been numerically illustrated in the present section. The numerical enunciates the procedure for evaluating the optimal release time of software and the security patch in the market. The actual fault count data of Tandem computers is obtained from the software literature [42] for

numerical purpose. In the data set, there are total of 100 faults that are discovered in 20 weeks of testing. The model parameters for the testing period are estimated using the nonlinear least square method, which is solved using the SPSS software. The estimated values of exponential SRGM are $a = 130.201$ and $b = 0.083$. The rest of the software parameters are assumed by making use of the experts option and previous studies. The assumed parameter values (cost values are in dollars) are given as:

$$\begin{aligned} r_1 = 0.2, r_2 = 0.3, r_3 = 0.4, C_1 = 70, C_2 = 5, C_3 = 30, \\ C_4 = 66, C_5 = 90, C_6 = 215, \\ R_0 = 0.9, w = 26 \text{ weeks} \end{aligned}$$

In the present study, an optimization software known as MAPLE has been applied to numerically solve and graphically interpret the optimization problem. After applying the values of all the parameters, following optimal results are obtained: optimal software time-to-market, $\tau^* = 21.86$ weeks and optimal patch time-to-market $\tau_1^* = 2.29$ weeks with the total minimum cost of $C(t)^* = 10061.61$ (in dollars). The convexity of the cost function is pictorially represented in Fig. 24.5. Further, the phase-wise faults discovered by testers and users are summarized in Table 24.6. From Table 24.6, it can be observed that testers are able to debug maximum number of faults (109 out of 130 faults) in the testing phase. However, the remaining untraced faults (21 faults) are being detected and reported by the customers to the testing team after the release of the software. Ten faults were detected during the warranty period and nine faults were detected postwarranty period.

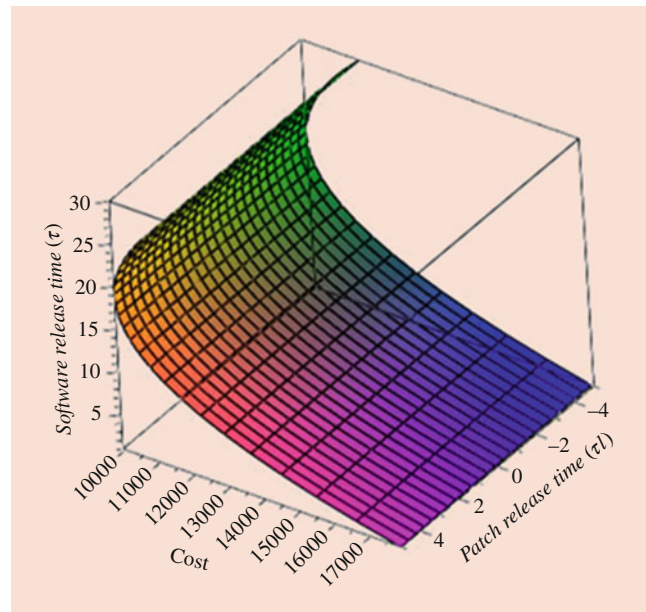


Fig. 24.5 Concavity plot for the minimization function

Table 24.6 Phase-wise quantification of vulnerabilities

Phase	Mean value function	No. of faults removed
Pre-release phase [0, τ]	$m(\tau^*)$	108.9 (109 approx.)
First patch release phase under warranty [τ , τ_1]	$m(\tau_1 - \tau^*)$	0.74 (1 approx.)
Postpatch release phase under Warranty [τ_1 , $\tau + w$]	$m(\tau + w - \tau_1^*)$	9.16 (9 approx.)
Postwarranty phase [τ , w, T_{lc}]	$m(T_{lc} - (\tau + w)^*)$	9.29 (9 approx.)

24.4.3 Optimal Patch Release Time for Both Faults and Vulnerabilities

In this section, the optimization model [43] for patch release time considers the debugging process of both faults and security vulnerabilities. When company develops different patches for faults and vulnerabilities, extra resources are required that increase both the development cost and time. Therefore, software developers tend to release a single patch that fixes both the “faults” and “vulnerabilities” of the software. In the present section, an optimal release time problem is studied with two optimal patch release approaches. According to first approach, both the faults and vulnerabilities are fixed by the developers via single patch. As the severity of vulnerabilities is much higher than the faults, the second approach considers the release of two patches where the first patch corrects both faults and vulnerabilities together while second patch is released by the testing team to specifically fixed vulnerabilities. Hence, two optimization problems are developed by keeping the cost minimization as an objective function.

Mathematical Model

The present software reliability growth model is based on the nonhomogenous Poisson process (NHPP). The hazard rate approach is applied for predicting the mean value function of total number of faults and vulnerabilities detected during the software life cycle. The differential equation of the mean value can be described as:

$$\frac{dm(t)}{dt} = \left(\frac{f(t)}{1 - F(t)} \right) (a \cdot p_i - m(t)) \quad \text{for } i = 1, 2 \tag{24.62}$$

where p_1 is the fraction of faults out of total a defects present in the software and p_2 is the fraction of vulnerabilities, also $p_1 + p_2 = 1$; $\frac{f(t)}{1 - F(t)}$ is the combined hazard rate of fault and vulnerability detection.

The above equation can be further solved using the initial value condition that at $t = 0$, $m(t) = 0$, $F(t) = 0$ to obtain the closed-form solution:

$$m(t) = a \cdot p_i F(t) \quad \text{for } i = 1, 2 \tag{24.63}$$

Two different patch release approaches are considered in this section. In the first, no clear distinction between the faults

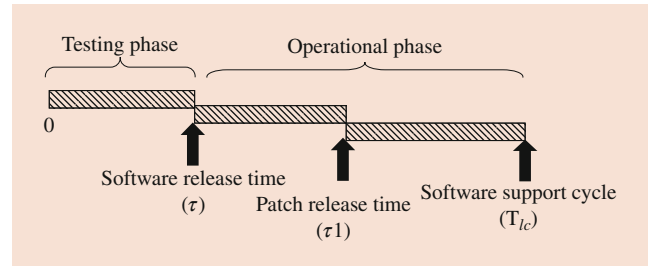


Fig. 24.6 Phase-wise description of software life cycle for faults and vulnerability discovery for approach 1

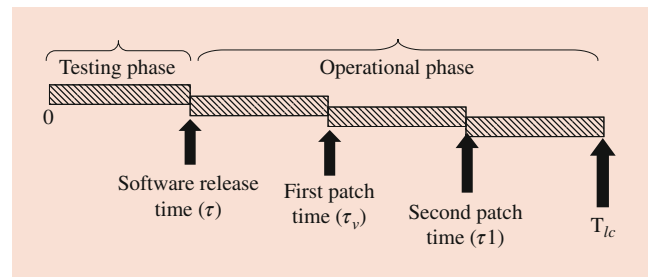


Fig. 24.7 Phase-wise description of software life cycle for faults and vulnerability discovery for approach 2

and vulnerabilities is employed with respect to criticality, discovery rate, and cost function. The pictorial representation of first approach is provided in Fig. 24.6. In the second approach, vulnerability prioritization is considered. According to this case, first patch specifically for vulnerability is released and then the fixing of faults is considered. The purpose of releasing the vulnerability patch is to make certain that the risks generated due to vulnerability exploitation are reduced on time. The phase-wise description of software life cycle for second approach is represented in Fig. 24.7.

Phase 1: Testing Phase [0, τ]

As mentioned in the previous section, testing team independently works to detect and debug defects present in the software. In this phase, debuggers assess every fault and vulnerability as equally severe. For both the approaches discussed above, the first phase has same mathematical interpretation. Thus, the expected number of faults and vulnerabilities identified during this phase can be described using the following expression:

$$m(\tau) = aF_1(\tau) \tag{24.64}$$

where $F_1(\tau)$ describes the fault distribution function of the testing team; τ is the release time of software in the market. It is further assumed that the faults detection phenomenon follows an exponential distribution pattern, that is,

$$m(\tau) = a(1 - e^{-b\tau}) \quad (24.65)$$

where b is the fault detection rate and a is the potential fault and vulnerability content present in the system.

Phase 2: Prepatch Release Phase $[\tau, \tau_1]$

Approach 1: When only single patch is released for fixing faults and vulnerabilities

After introducing the software in the market, the work of critical identification of faults and vulnerabilities is left with customers. However, the efficiency of users is much lower as compared to the professional testers. Moreover, the seriousness and criticality of vulnerability is much intensive than any other fault. Also, the cost incurred in the discovery and correction of vulnerability is quite higher as compared to the cost employed for debugging faults. Therefore, for developer's convenience, from this interval the defect distinction process starts for evaluating the debugging cost for faults and vulnerabilities separately. The total number of faults discovered and removed in this interval is represented as:

$$m_f(\tau_1 - \tau) = a \cdot p_1 \cdot (1 - F_1(\tau)) \cdot F_2(\tau_1 - \tau) \quad (24.66)$$

where $a \cdot p_1 \cdot (1 - F_1(\tau))$ is defined as the proportion of faults remained to debug from the previous phase; $F_2(\tau_1 - \tau)$ is fault detection function of the users. It is further assumed that the faults detection phenomenon follows an exponential distribution pattern, that is,

$$m_f(\tau_1 - \tau) = a \cdot p_1 \cdot e^{-b\tau} \cdot (1 - e^{-b \cdot r_1 \cdot (\tau_1 - \tau)}) \quad (24.67)$$

where $b \cdot r_1$ ($0 < r_1 \leq 1$) is the fault detection rate of the users which is some fraction of the tester's fault detection rate. In similar lines, the expected vulnerabilities discovered and corrected in this interval are defined as:

$$\Omega(\tau_1 - \tau) = a \cdot p_2 \cdot (1 - F_1(\tau)) \cdot F_3(\tau_1 - \tau) \quad (24.68)$$

where $a \cdot p_2 \cdot (1 - F_1(\tau))$ is defined as the proportion of vulnerability remained to debug from the previous phase; $F_3(\tau_1 - \tau)$ is vulnerability detection function of the users. If vulnerability discovery phenomenon follows exponential distribution function, then

$$\Omega(\tau_1 - \tau) = a \cdot p_2 \cdot e^{-b\tau} \cdot (1 - e^{-b \cdot r_2 \cdot (\tau_1 - \tau)}) \quad (24.69)$$

Approach 2: When two different patches are released for fixing vulnerabilities and both faults and vulnerabilities, respectively

According to this scenario, if developers encounter any vulnerability in the list of reported defects then they construct a patch for vulnerabilities first because they have more severe impact than any other faults. In addition, a second patch is released for correcting both vulnerabilities and faults collectively after releasing the vulnerability patch first. At time point τ_v , the patch for reported vulnerabilities during the time interval $[\tau, \tau_v]$ is released. The expected number of vulnerabilities reported during this interval becomes:

$$\Omega(\tau_v - \tau) = a \cdot p_2 \cdot (1 - F_1(\tau)) \cdot F_2(\tau_v - \tau) \quad (24.70)$$

where $a \cdot p_2 \cdot (1 - F_1(\tau))$ is the proportion of vulnerabilities that remained undetected in the testing phase; $F_2(\tau_v - \tau)$ is the vulnerability detection function of the users. Now, as the vulnerability discovery follows an exponential distribution function, the mean value function of vulnerability discovery can be described as:

$$\Omega(\tau_v - \tau) = a \cdot p_2 \cdot e^{-b\tau} \cdot (1 - e^{-b \cdot r_2 \cdot (\tau_v - \tau)}) \quad (24.71)$$

Further, a second patch is released at time-point τ_1 to fix faults and vulnerabilities simultaneously. The expected number of vulnerabilities reported by the users in the interval $[\tau_v, \tau_1]$ until the second patch is given as:

$$\begin{aligned} \Omega(\tau_1 - \tau_v) &= a \cdot p_2 \cdot (1 - F_1(\tau)) \cdot (1 - F_2(\tau_v - \tau)) \\ &\quad \cdot F_3(\tau_1 - \tau_v) \end{aligned} \quad (24.72)$$

where $a \cdot p_2 \cdot (1 - F_1(\tau)) \cdot (1 - F_2(\tau_v - \tau))$ is defined as the proportion of vulnerabilities that remain undetected by time τ_v ; $F_3(\tau_1 - \tau_v)$ is the cumulative distribution function of vulnerability discovery. When vulnerability discovery phenomenon follows exponential distribution function, then

$$\Omega(\tau_1 - \tau_v) = a \cdot p_2 \cdot e^{-b\tau} \cdot e^{-b \cdot r_2 \cdot (\tau_v - \tau)} \cdot (1 - e^{-b \cdot r_2 \cdot (\tau_1 - \tau_v)}) \quad (24.73)$$

Similarly, the expected number of faults identified and reported by the users in the entire interval $[\tau, \tau_1]$ until the patch for faults is released is given as:

$$m_f(\tau_1 - \tau) = a \cdot p_1 \cdot (1 - F_1(\tau)) \cdot F_4(\tau_1 - \tau) \quad (24.74)$$

where $a \cdot p_1 \cdot (1 - F_1(\tau))$ is the proportion of faults that remain unidentified in the previous phase; $F_4(\tau_1 - \tau)$ is the cumulative distribution function of fault detection by the users in the second phase. If fault detection phenomenon

follows exponential function, then expected number of faults identified during this phase becomes:

$$m_f(\tau_1 - \tau) = a \cdot p_1 \cdot e^{-b\tau} \cdot (1 - e^{-b \cdot r_1 \cdot (\tau_1 - \tau)}) \quad (24.75)$$

Phase 3: Postpatch Release Phase $[\tau_1, T_{lc}]$

Approach 1: When only single patch is released for fixing faults and vulnerabilities

After the patch is delivered to the users, they install it to fix the bugs and vulnerabilities, which were identified and reported by them. However, not all defects will be fixed by the single patch. Therefore, faults and vulnerability discovery phenomenon continues even after the release of a patch. Developers assist their customers for correcting the bugs and security vulnerabilities till the software lifecycle. Thus, the expected number of faults detected by the users during postrelease phase becomes:

$$m_f(T_{lc} - \tau_1) = a \cdot p_1 \cdot (1 - F_1(\tau)) \cdot (1 - F_2(\tau_1 - \tau)) \cdot F_4(T_{lc} - \tau_1) \quad (24.76)$$

Further, the total number of vulnerabilities discovered and removed during this phase is expressed as:

$$\Omega(T_{lc} - \tau_1) = a \cdot p_2 \cdot (1 - F_1(\tau)) \cdot (1 - F_3(\tau_1 - \tau)) \cdot F_5(T_{lc} - \tau_1) \quad (24.77)$$

where $a \cdot p_1 \cdot (1 - F_1(\tau)) \cdot (1 - F_2(\tau_1 - \tau))$ and $a \cdot p_1 \cdot (1 - F_1(\tau)) \cdot (1 - F_3(\tau_1 - \tau))$ are the residual faults and vulnerabilities, respectively, that remain undetected in the previous period; $F_4(T_{lc} - \tau_1)$ is the fault detection function and $F_5(T_{lc} - \tau_1)$ is the vulnerability discovery function in duration $[\tau_1, T_{lc}]$. For exponentially discovery distribution, the expected number of faults and vulnerabilities detection, respectively, becomes:

$$m_f(T_{lc} - \tau_1) = a \cdot p_1 \cdot e^{-b\tau} \cdot e^{-b \cdot r_1 \cdot (\tau_1 - \tau)} \cdot (1 - e^{-b \cdot r_1 \cdot (T_{lc} - \tau_1)}) \quad (24.78)$$

$$\Omega(T_{lc} - \tau_1) = a \cdot p_2 \cdot e^{-b\tau} \cdot e^{-b \cdot r_2 \cdot (\tau_1 - \tau)} \cdot (1 - e^{-b \cdot r_2 \cdot (T_{lc} - \tau_1)}) \quad (24.79)$$

Approach 2: When two different patches are released for fixing vulnerabilities and both faults and vulnerabilities, respectively

After the release of two patches at time-point τ_v and τ_1 , the fault and vulnerability discovery process continues until the software life cycle ends at time T_{lc} . Therefore, the expected number of faults identified in the last phase is given as:

$$m_f(T_{lc} - \tau_1) = a \cdot p_1 \cdot (1 - F_1(\tau)) \cdot (1 - F_4(\tau_1 - \tau)) \cdot F_5(T_{lc} - \tau_1) \quad (24.80)$$

Similarly, the expected number of vulnerabilities discovered is represented as:

$$\Omega(T_{lc} - \tau_1) = a \cdot p_1 \cdot (1 - F_1(\tau)) \cdot (1 - F_2(\tau_v - \tau)) \cdot (1 - F_3(\tau_1 - \tau_v)) \cdot F_6(T_{lc} - \tau_1) \quad (24.81)$$

When detection distribution follows exponential function, then

$$m_f(T_{lc} - \tau_1) = a \cdot p_1 \cdot e^{-b\tau} \cdot e^{-b \cdot r_1 \cdot (\tau_1 - \tau)} \cdot (1 - e^{-b \cdot r_1 \cdot (T_{lc} - \tau_1)}) \quad (24.82)$$

$$\Omega(T_{lc} - \tau_1) = a \cdot p_2 \cdot e^{-b\tau} \cdot e^{-b \cdot r_2 \cdot (\tau_v - \tau)} \cdot e^{-b \cdot r_2 \cdot (\tau_1 - \tau_v)} \cdot (1 - e^{-b \cdot r_2 \cdot (T_{lc} - \tau_1)}) \quad (24.83)$$

Cost Modeling

The cost functions associated with the testing and debugging of faults and vulnerabilities in the software are listed in Table 24.7.

Thus, the total cost can be obtained by adding all the cost components become:

$$\text{Total_Cost, } C(t) = C_{\text{testing_phase}}(t) + C_{[0, \tau]}(t) + C_{[\tau, \tau_1]}(t) + C_{[\tau_1, T_{lc}]}(t) \quad (24.84)$$

The optimization problem with an aim of minimizing the overall cost function is defined as:

Approach 1: Single-patch model

Minimize

$$C^1(t) = C_1 \cdot \tau + C_2 \cdot a (1 - e^{-b\tau}) + C_3 \cdot a p_1 (e^{-b\tau} (1 - e^{-b r_1 (\tau_1 - \tau)}) + C_4 \cdot a p_2 (e^{-b\tau} (1 - e^{-b r_2 (\tau_1 - \tau)}) + C_5 \cdot a p_1 (e^{-b\tau} (e^{-b r_1 (\tau_1 - \tau)}) (1 - e^{-b r_1 (T_{lc} - \tau_1)}) + C_6 \cdot a p_2 (e^{-b\tau} (e^{-b r_2 (\tau_1 - \tau)}) (1 - e^{-b r_2 (T_{lc} - \tau_1)})) \quad (24.85)$$

Table 24.7 Components of different cost functions

Cost components	Description	Cost function
Testing cost	Testing cost refers to per unit testing cost and is associated with the tester’s activities such as test planning, test case generation, test execution, and analysis of testing results.	$C_{\text{testing}}(t) = C_1 \tau$
Faults debugging cost during testing phase	This cost comprises direct cost associated with identifying and fixing the bugs	$C_{[0, \tau]}(t) = C_2 m(\tau)$
Faults debugging cost during prepatch release phase	Debugging cost during this phase is the cost of debugging of the faults and vulnerabilities detected and reported by the users	Approach 1: $C_{[\tau, \tau_1]}(t) = C_3 m_f (\tau_1 - \tau) + C_4 \Omega (\tau_1 - \tau)$ Approach 2: $C_{[\tau, \tau_1]}(t) = C_3 \Omega (\tau_v - \tau) + C_4 \Omega (\tau_1 - \tau_v) + C_5 m_f (\tau_1 - \tau)$ where $\tau < \tau_v < \tau_1$
Faults debugging cost during postpatch release period	This cost includes the debugging cost in operational phase after patch has been released for the faults and vulnerabilities reported by the users	Approach 1: $C_{[\tau_1, T_{lc}]}(t) = C_5 m_f (T_{lc} - \tau_1) + C_6 \Omega (T_{lc} - \tau_1)$ Approach 2: $C_{[\tau_1, T_{lc}]}(t) = C_6 m_f (T_{lc} - \tau_1) + C_7 \Omega (T_{lc} - \tau_1)$

Approach 2: Two-patch model
Minimize

$$\begin{aligned}
 C^2(t) = & C_1 \cdot \tau + C_2 \cdot a (1 - e^{-b\tau}) + C_3 \cdot ap_2 (e^{-b\tau} \\
 & (1 - e^{-br_2(r_v-\tau)}) + C_4 \cdot ap_2 (e^{-b\tau}) (e^{-br_2(r_v-\tau)}) \\
 & (1 - e^{(-br_2(\tau_1-\tau_v))}) + C_5 \cdot ap_1 (e^{-b\tau}) (1 - e^{-br_1(\tau_1-\tau)}) \\
 & + C_6 \cdot ap_1 (e^{-b\tau}) (e^{-br_1(r_1-\tau)}) (1 - e^{-br_1(T_{lc}-\tau_1)}) \\
 & + C_7 \cdot ap_2 (e^{-b\tau}) (e^{-br_2(r_v-\tau)}) (e^{(-br_2(\tau_1-\tau_v))}) \\
 & (1 - e^{-br_2(T_{lc}-\tau_1)})
 \end{aligned}
 \tag{24.86}$$

Numerical Example

To understand the practical application of the above-discussed model, a numerical example is provided. The past data set of fault count consisting of 535 defects that were identified during the testing of 109 days was obtained from Tohma et al. [44]. It is considered that the data set contains both faults and vulnerabilities. The mean value function for exponential model is $m(t) = a \cdot p_1 \cdot (1 - \exp(-b \cdot t))$ for faults and $m(t) = a \cdot p_2 \cdot (1 - \exp(-b \cdot t))$ for vulnerabilities. The estimated values of the parameters of the exponential model using the SPSS software are: $a = 713.889$ and $b = 0.015$. Further, the remaining parameter values are assumed based on the experts opinion, that is, $r_1 = 0.5$, $r_2 = 0.3$, $p_1 = 0.9$, $p_2 = 0.1$, and $T_{lc} = 1000$.

The objective of the optimization problem is to evaluate the optimal time-to-market the new software and optimal time to release the patch for fixing the vulnerabilities and faults. The cost minimization problem is solved using the computational software known as MAPLE using the following cost parameters value (in dollars):

$$C_1 = 350, C_2 = 260, C_3 = 300, C_4 = 2400, C_5 = 310, \\
 C_6 = 2300 \text{ (For Approach 1)}$$

$$C_1 = 350, C_2 = 260, C_3 = 2400, C_4 = 2400,$$

$$C_5 = 300, C_6 = 310, C_7 = 2300 \text{ (For Approach 2)}$$

The optimal results obtained using the above-mentioned values of the parameter set are summarized in Table 24.8. From the results listed in the Table 24.8, it can be comprehend that for the both the approaches whether releasing a single patch or two patches, the optimal time to release a software will be same, that is, the software should be released after approximately 114 days of testing. Nevertheless, according to the actual observation of the debug data, the testing was done only for 109 days in which 535 defects were found. It indicates that the developer should extend the testing for 5 more days in order to debug more faults and vulnerabilities at a minimum cost.

Further, the phase-wise description of number of faults and vulnerabilities debugged is summarized in Table 24.9. From Table 24.9, it can be observed that the number of defects detected/debugged in the testing phase is 584.6 (approximately 585). If testing is continued for extra 5 days, then 50 more defects can be debugged in the testing phase. Also, during operational phase, the number of faults detected in prepatch release phase are 74.12 and number of vulnerabilities discovered/patched are 5.89 when only single patch is released. Similarly, when two patches are released, then the total vulnerabilities discovered until second patch is released are 5.88. Additionally, the total defects debugged in the post patch release in both the approaches are 48.86.

24.4.4 Optimal Patch Release Time for Vulnerable Software Systems

In this section, the debugging process of vulnerabilities in the operational phase is explicitly considered. The optimal patch time-to-market for fixing the software vulnerabilities

Table 24.8 Optimal release time results

Patch policy	Optimal release time	Optimal patch release time		Optimal cost (in dollars)
Approach 1	113.9	135.15		239606.9
		First patch	Second patch	
Approach 2	113.9	75.03	60.12	239606.9

Table 24.9 Phase-wise description of defects using different approaches

Single-patch optimization problem			Two-patch optimization problem		
Phase	Mean value function	No. of defects removed	Phase	Mean value function	No. of defects removed
[0, τ]	$m(\tau)$	584.6	[0, τ]	$m(\tau)$	584.6
[τ, τ ₁]	$m_f(\tau_1 - \tau)$	74.12	[τ, τ _v]	$\Omega(\tau_v - \tau)$	3.7
	$\Omega(\tau_1 - \tau)$	5.89	[τ _v , τ ₁]	$\Omega(\tau_1 - \tau_v)$	2.18
[τ ₁ , T _{lc}]	$m_f(T_{lc} - \tau_1)$	42.07	[τ, τ ₁]	$m_f(\tau_1 - \tau)$	74.12
	$\Omega(T_{lc} - \tau_1)$	6.79	[τ ₁ , T _{lc}]	$m_f(T_{lc} - \tau_1)$	42.07
				$\Omega(T_{lc} - \tau_1)$	6.79

whose severity is much greater than the faults is determined. It is assumed that the software is released at time-point 0. The objective of the study is to determine the optimal patch release time τ that minimizes the total cost incurred by developer in vulnerability discovery process. Further, the vulnerability discovery process in the operational phase is divided into parts: prepatch release phase $[0, \tau]$ and postpatch release phase $[\tau, T_{lc}]$. Also, the time instant at which software vendor releases the patch is also the testing termination time. It is also assumed that the cost of patching is negligible and does not introduce any new vulnerability in the system.

Mathematical Modeling

The mathematical expression for the vulnerability discovery process in the operational phase is described in this subsection. The pictorial representation of phase-wise description of software life cycle is given in Fig. 24.8. When software is in market, the vulnerabilities are identified by both the testers and the users. Users on the other hand may belong to either category of *white hat* or *black hat* depending on their intention of discovering vulnerabilities. Black hat are unethical hackers who attempt to identify security loopholes in order to exploit the system, whereas white hat discover vulnerabilities with the permission of the company. They act as reporters who inform about the security weaknesses to the appropriate authority in order to fix them.

Phase 1: Prepatch Release Phase [0, τ]

It is considered that software product is released in the market at time 0. Even after the release, some security breaches are still present in the system. Therefore, in operational phase, vulnerabilities are being identified by the testers and users (black hat and white hat), that is,

$$\Omega(\tau) = \Omega_{\text{tester}}(\tau) + \Omega_{\text{black_hat}}(\tau) + \Omega_{\text{white_hat}}(\tau) \tag{24.87}$$

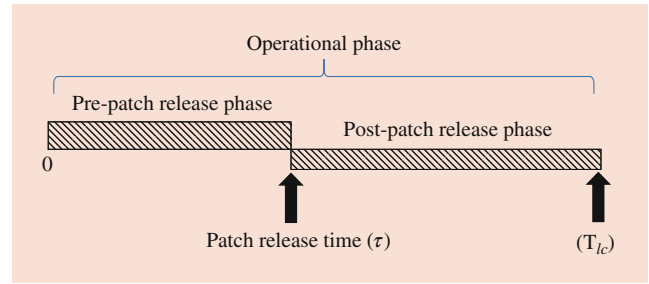


Fig. 24.8 Phase-wise description of software life cycle for vulnerability discovery

Besides, it is assumed that the some fraction, say λ_1 , of the total vulnerabilities is discovered by the testing team during this phase and identified by the testing team during field-testing. The λ_2 fraction of undetected vulnerabilities is being exploited by the black hat users and the remaining, that is, $\lambda_3 = (1 - \lambda_1 - \lambda_2)$ proportion of the undiscovered bugs, are detected by the customers who then immediately report it to the developers for correcting it. Thus, the total vulnerability discovery during this phase becomes:

$$\Omega(\tau) = \lambda_1 \bar{V}F_{\text{tester}}(\tau) + \lambda_2 \bar{V}F_{\text{black_hat}}(\tau) + \lambda_3 \bar{V}F_{\text{white_hat}}(\tau) \tag{24.88}$$

where

$F_{\text{tester}}(\tau) = (1 - e^{-b\tau})$ is the exponential distribution function for vulnerability discovery by testers in interval $[0, \tau]$

$F_{\text{black_hat}}(\tau) = (1 - e^{-br\tau})$ is the exponential distribution function for vulnerability discovery by hackers or black hat users

$F_{\text{white_hat}}(\tau) = (1 - e^{-bs\tau})$ is the exponential distribution function for vulnerability discovery by white hat users

Moreover, the vulnerability discovery rate of black-hat ($b \cdot r$) and white hat ($b \cdot s$) users is some fraction of the vulnerability discovery rate of the testing team.

Phase 2: Postpatch Release Phase [τ, T_{lc}]

At time τ , the developers release the patch for the discovered vulnerabilities and at this time point testers stop the vulnerability discovery process. Therefore, during postpatch release phase, only users either black hat or white hat identify the vulnerabilities in the software system. The remaining number of vulnerabilities that were undetected in previous phase is discovered during this period. Let λ_4 proportion of undetected vulnerabilities is identified by the hackers and the remaining proportion of vulnerabilities, that is, $1 - \lambda_4 = \lambda_5$, is detected by the white hat users. Thus, the total number of vulnerabilities discovered in this phase is given by:

$$\Omega(T_{lc} - \tau) = \lambda_4 (\bar{V} - \Omega(\tau)) F_{\text{black_hat}}(T_{lc} - \tau) + \lambda_5 (\bar{V} - \Omega(\tau)) F_{\text{white_hat}}(T_{lc} - \tau) \tag{24.89}$$

where $(\bar{V} - \Omega(\tau))$ is the undiscovered vulnerabilities from phase 1 that are identified by the users in this phase and $\lambda_4 + \lambda_5 = 1$. If vulnerability discovery process follows exponential distribution function, then

$$\begin{aligned} \Omega(T_{lc} - \tau) = & \lambda_4 (\bar{V} - \lambda_1 \bar{V} (1 - e^{-b r \tau}) \\ & - \lambda_2 \bar{V} (1 - e^{-b r \tau}) - \lambda_3 \bar{V} (1 - e^{-b s \tau})) (1 - e^{-b r (T_{lc} - \tau)}) \\ & + \lambda_5 (\bar{V} - \lambda_1 \bar{V} (1 - e^{-b r \tau}) - \lambda_2 \bar{V} (1 - e^{-b r \tau}) \\ & - \lambda_3 \bar{V} (1 - e^{-b s \tau})) (1 - e^{-b s (T_{lc} - \tau)}) \end{aligned} \tag{24.90}$$

Cost Modeling

The cost functions associated with the testing and debugging of vulnerabilities in the software system are listed in Table 24.10.

Thus, the objective function can be obtained by adding all the cost components:

$$\begin{aligned} \text{Minimize Total_Cost, } C(t) = & C_{\text{testing_phase}}(t) + C_{[0, \tau]}(t) \\ & + C_{[\tau, T_{lc}]}(t) \end{aligned} \tag{24.91}$$

Numerical Example

The vulnerability data for numerical illustration purpose is collected from National Vulnerability Database (NVD), the Mitre Corporation website, and ICAT database. The testing and debugging data of an operating system Windows XP (SP2 Professional) is obtained for the present study. In this data set, execution time is reported in terms of months and vulnerabilities in terms of cumulative vulnerabilities at the end of each month. The data set consists of 17 vulnerabilities, which were reported from January 2007 to 2013. Statistical Package for Social Science (SPSS) tool has been used for estimating the parameter values. The estimated values of the parameters are $\bar{V} = 17.902$ and $b = 0.284$.

Furthermore, the vulnerability discovery rate of hacker and white hat users is considered some fraction of the tester’s vulnerability discovery rate, that is, the detection rate of black hat and white hat users are $b \cdot r$ and $b \cdot s$ ($0 < r, s \leq 1$), respectively. The remaining parameter values (cost parameters are in dollars) are assumed based on the expert’s opinion, that is,

$$\begin{aligned} r = 0.1136, \quad s = 0.0568, \quad C_1 = 50, \quad C_2 = 100, \quad C_3 = 250, \\ C_4 = 150, \quad C_5 = 250, \quad C_6 = 150, \\ \lambda_1 = 0.5, \quad \lambda_2 = 0.3, \quad \lambda_3 = 0.2, \quad \lambda_4 = 0.6, \quad \lambda_5 = 0.4, \\ T_{lc} = 50 \end{aligned}$$

The values of the parameters are substituted in the objective function to evaluate the optimal time-to market the patch, which minimizes the total cost function. The optimization problem is solved using MAPLE software and calculated value of optimal patch release time as $\tau^* = 3.320$, that is, software vendors should release the software vulnerability patch after 3 months (approximately) from the

Table 24.10 Components of different cost functions

Cost components	Description	Cost function
Testing cost	Testing cost refers to per unit testing cost and is associated with the tester’s activities such as test planning, test case generation, test execution, and analysis of testing results	$C_{\text{testing}}(t) = C_1 \tau$
Vulnerability discovery cost during prepatch release phase	This cost function includes the efforts expended on discovering the vulnerabilities by the testers and white hat users. It also includes the loss company suffer due the vulnerabilities identified by the black hat users	$C_{[0, \tau]}(t) = C_2 \Omega_{\text{tester}}(\tau) + C_3 \Omega_{\text{black_hat}}(\tau) + C_4 \Omega_{\text{white_hat}}(\tau)$
Vulnerability discovery cost during postpatch release period	This includes the cost of vulnerabilities discovery by the users after patch release	$C_{[\tau, T_{lc}]}(t) = C_5 \Omega_{\text{black_hat}}(T_{lc} - \tau) + C_6 \Omega_{\text{white_hat}}(T_{lc} - \tau)$

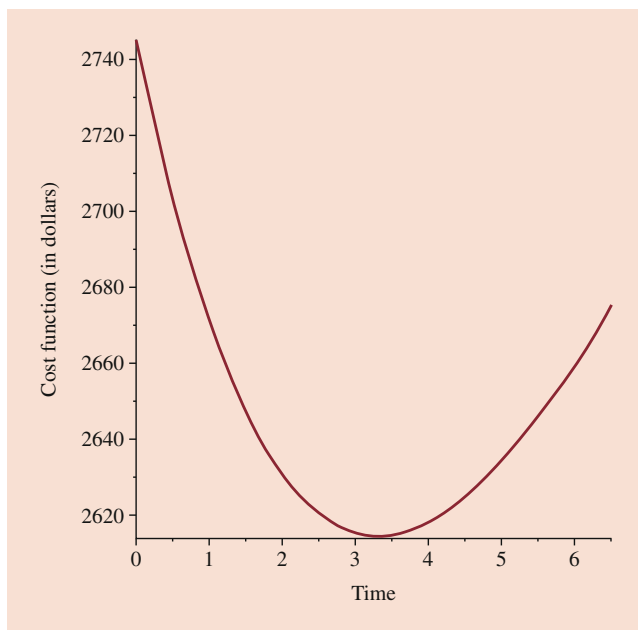


Fig. 24.9 Convexity plot for the cost function

release of the software and corresponding minimum cost is $C^*(t) = 2614.76$ (in dollars). The convexity of the objective function is established using the graph as shown in Fig. 24.9.

24.5 Concluding Remarks

The present study focuses on predicting the vulnerability discovery and optimal patching phenomenon of software products. The first part of the chapter examines the vulnerability discovery process by reviewing the mathematical models that examines the discovery pattern of security weaknesses. The role of software users, resource consumption, and coverage function on the defect detection process are analyzed. The actual vulnerability data of the software products were obtained to validate the models. The second part of the chapter centered on determining the optimal time to release the software and security patch in the market. Three different cost models are formulated by considering different scenarios. The results obtained in the current research can help managers in assessing the vulnerability discovery growth rate and optimal patch management practices.

References

1. Yamada, S.: *Software Reliability Modeling: Fundamentals and Applications*, vol. 5. Springer, Tokyo (2014)
2. Kapur, P.K., Kumar, S., Garg, R.B.: *Contributions to Hardware and Software Reliability*, vol. 3. World Scientific (1999)
3. Crook, R., Ince, D., Lin, L.: B. Nuseibeh: security requirements engineering: when anti-requirements hit the fan. In: Proc. IEEE Joint International Conference on Requirements Engineering, pp. 203–205 (2002)
4. Younis, A., Malaiya, Y.K., Ray, I.: Assessing vulnerability exploitability risk using software properties. *Softw. Qual. J.* **24**(1), 159–202 (2016)
5. Ghaffarian, S.M., Shahriari, H.R.: Software vulnerability analysis and discovery using machine-learning and data-mining techniques: a survey. *ACM Comput. Surv. (CSUR)*, **50**(4), 56 (2017)
6. Kapur, P.K., Sachdeva, N., Khatri, S.K.: Vulnerability discovery modeling. In: Proc. International Conference on Quality, Reliability, Infocom Technology and Industrial Technology Management, pp. 34–54 (2015)
7. Computer Emergency Response Team, <https://www.sei.cmu.edu/about/divisions/cert/index.cfm>
8. National Vulnerability Database, <https://nvd.nist.gov/>
9. Joh, H., Malaiya, Y.K.: Periodicity in software vulnerability discovery, patching and exploitation. *Int. J. Inf. Secur.* **16**(6), 673–690 (2017)
10. Anderson, R.: Why information security is hard—an economic perspective. In: Proc 17th Annual Computer Security Applications Conference, ACSAC 2001, pp. 358–365 (2001, December)
11. Rescorla, E.: Security holes. Who cares? In: SSYM'03: Proceedings of the 12th Conference on USENIX Security Symposium, pp. 75–90. USENIX Association, Berkeley (2003)
12. Alhazmi, O.H., Malaiya, Y.K.: Modeling the vulnerability discovery process. In: Proc. 16th IEEE International Symposium on Software Reliability Engineering (ISSRE'05), p. 10. IEEE (2005)
13. Joh, H., Kim, J., Malaiya, Y.K.: Vulnerability discovery modeling using Weibull distribution. In: Proc. 19th International Symposium on Software Reliability Engineering. ISSRE 2008, pp. 299–300. IEEE (2008)
14. Kapur, P.K., Yadavali, V.S., Shrivastava, A.K.: A comparative study of vulnerability discovery modeling and software reliability growth modeling. In: Proc. 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), pp. 246–251. IEEE (2015)
15. Kim, J., Malaiya, Y.K., Ray, I.: Vulnerability discovery in multi-version software systems. In: Proc. 10th IEEE High Assurance Systems Engineering Symposium, HASE '07, pp. 141–148 (2007)
16. Ozment, A.: Improving vulnerability discovery models. In: Proc. Proceedings of the 2007 ACM Workshop on Quality of Protection, pp. 6–11. ACM (2007)
17. Sutton, M., Greene, A., Amini, P.: *Fuzzing: Brute Force Vulnerability Discovery*. Pearson Education (2007)
18. Algarni, A., Malaiya, Y.K.: Software vulnerability markets: discoverers and buyers. *Int. J. Comput. Inf. Sci. Eng.* **8**(3), 71–81 (2014)
19. Nguyen, V. H.: Empirical methods for evaluating vulnerability models. Ph.D. Thesis, University of Trento (2014)
20. Zhu, X., Cao, C., Zhang, J.: Vulnerability severity prediction and risk metric modeling for software. *J. Appl. Intell.* **47**(3), 828–836 (2017)
21. Pham, H., Zhang, X.: Software release policies with gain in reliability justifying the costs. *Ann. Softw. Eng.* **8**(1–4), 147–166 (1999)
22. Inoue, S., Yamada, S.: Optimal software release policy with change-point. In: Proc. IEEE International Conference on Industrial Engineering and Engineering Management, IEEM, pp. 531–535. IEEE (2008)
23. Kapur, P.K., Pham, H., Gupta, A., Jha, P.C.: *Software Reliability Assessment with OR Applications*, p. 364. Springer, London (2011)
24. Lai, R., Garg, M., Kapur, P.K., Liu, S.: A study of when to release a software product from the perspective of software reliability models. *JSW.* **6**(4), 651–661 (2011)
25. Kapur, P.K., Singh, V.B., Singh, O., Singh, J.N.: Software release time based on different multi-attribute utility functions. *Int. J. Reliab. Qual. Saf. Eng.* **20**(04), 1350012 (2013)

26. Kapur, P.K., Khatri, S.K., Tickoo, A., Shatnawi, O.: Release time determination depending on number of test runs using multi attribute utility theory. *Int. J. Syst. Assur. Eng. Manag.* **5**(2), 186–194 (2014)
27. Kapur, P.K., Panwar, S., Singh, O., Kumar, V.: Joint release and testing stop time policy with testing-effort and change point. In: *Risk Based Technologies*, pp. 209–222. Springer, Singapore (2019)
28. Beattie, S., Arnold, S., Cowan, C., Wagle, P., Wright, C., Shostack, A.: Timing the application of security patches for optimal uptime. In: *Proc. LISA '02: Sixteenth Systems Administration Conference*, vol. 2, pp. 233–242 (2002)
29. Telang, R., Wattal, S.: An empirical analysis of the impact of software vulnerability announcements on firm stock price. *IEEE Trans. Softw. Eng.* **33**(8), 544–557 (2007)
30. Arora, A., Telang, R., Xu, H.: Optimal policy for software vulnerability disclosure. *Manag. Sci.* **54**(4), 642–656 (2008)
31. Arora, A., Forman, C., Nandkumar, A., Telang, R.: Competition and patching of security vulnerabilities: an empirical analysis. *Inf. Econ. Policy.* **22**(2), 164–177 (2010)
32. Kansal, Y., Kapur, P.K., Kumar, U., Kumar, D.: User-dependent vulnerability discovery model and its interdisciplinary nature. *Life Cycle Reliability and Safety Engineering.* **6**(1), 23–29 (2017)
33. Zimmerman, R.: Decision-making and the vulnerability of interdependent critical infrastructure. In: *Proc. IEEE International Conference on Systems, Man and Cybernetics*, vol. 5, pp. 4059–4063. IEEE (2004)
34. Kenny, G.Q.: Estimating defects in a commercial software during operational use. *IEEE Trans. Reliab.* **42**(1), 107–115 (1993)
35. Bass, F.M.: A new-product growth model for consumer durables. *Manag. Sci.* **15**(5), 215–227 (1969)
36. Kansal, Y., Kapur, P.K., Kumar, U., Kumar, D.: Effort and coverage dependent vulnerability discovery modeling. In: *Proc. 2nd International Conference on Telecommunication and Networks (TELNET)*, pp. 1–6. IEEE (2017)
37. Pham, H.: A generalized fault-detection software reliability model subject to random operating environments. *Vietnam J. Comput. Sci.* **3**(3), 145–150 (2016)
38. Kansal, Y., Kapur, P.K., Kumar, U.: Coverage-based vulnerability discovery modeling to optimize disclosure time using multiattribute approach. *Qual. Reliab. Eng. Int.* **35**(1), 62–73 (2019)
39. Common Vulnerability Exposure repository, <https://www.cvedetails.com>
40. Yamada, S., Tamura, Y.: *OSS Reliability Measurement and Assessment*. Springer International Publishing (2016)
41. Kansal, Y., Singh, G., Kumar, U., Kapur, P.K.: Optimal release and patching time of software with warranty. *Int. J. Syst. Assur. Eng. Manag.* **7**(4), 462–468 (2016)
42. Wood, A.: Predicting software reliability. *Computer.* **29**(11), 69–77 (1996)
43. Kansal, Y., Kumar, U., Kumar, D., Kapur, P.K.: Fixing of faults and vulnerabilities via single patch. In: *Quality, IT and Business Operations*, pp. 175–190. Springer, Singapore (2018)
44. Tohma, Y., Yamano, H., Ohba, M., Jacoby, R.: Parameter estimation of the hyper-geometric distribution model for real test/debug data. In: *Proc. International Symposium on Software Reliability Engineering*, pp. 28–34. IEEE (1991)



P. K. Kapur is the Director of Amity Center for Interdisciplinary Research, Amity University, Noida. He is the Former Dean of the Faculty of Mathematical Sciences, and the Former Head of the Department of Operational Research, University of Delhi. His vast research experience in the areas of Software Reliability, Optimization, Technology Management, Hardware Reliability, Multi-Criteria Decision Making (MCDM), and other areas of management, is illustrated through his work with nearly 450 research paper publications in top international and national journals/proceedings of repute, and supervision of over 45 Ph.D. & 25 M.Phil. scholars. He is also editor-in-chief of the *International Journal of Systems Assurance Engineering and Management (IJSAEM)*, Springer.



Saurabh Panwar is a Ph.D. in Operations Research from the Department of Operational Research, University of Delhi, Delhi, India. He is associated with the Department of Finance and Business Economics, South Campus, University of Delhi and Delhi School of Professional Studies and Research, Guru Gobind Singh Indraprastha University, Delhi, as a Guest Faculty. He is also an Associate Editor of the *International Journal of Systems Assurance Engineering and Management (IJSAEM)*, Springer. He was judged as a *Young Promising Researcher* of international repute by the President of SREQOM in 2018. His research interests include mathematical modeling, new product development, technological forecasting, software reliability, and optimization.



Contents

25.1	A Generalized NHPP Software Reliability Model . . .	483
25.2	Generalized Random Field Environment (RFE) Model	483
25.3	RFE Software Reliability Models	484
25.3.1	γ -RFE Model	484
25.3.2	β -RFE Model	485
25.4	Parameter Estimation	486
25.4.1	Maximum Likelihood Estimation (MLE)	486
25.4.2	Mean-Value Function Fits	487
25.4.3	Software Reliability	488
25.4.4	Confidence Interval	488
25.4.5	Concluding and Remarks	491
25.5	A RFE Model with Vtub-Shaped Fault-Detection Rate	492
25.5.1	Model Criteria	492
25.5.2	Model Analysis	492
References	493

testing and field operation environments, this generalized model considers the random environmental effects on software reliability. Based on the generalized RFE model, Sect. 25.4 describes two specific RFE reliability models, the γ -RFE and β -RFE models, for predicting software reliability in field environments. Section 25.5 illustrates the models using telecommunication software failure data. Some further considerations based on the generalized software reliability model are also discussed.

Keywords

Reliability prediction · Software testing · Software reliability · Software development process · Software failure · Model criteria

Abstract

After a brief overview of existing models in software reliability in Sects. 25.1 and 25.2 discusses a generalized nonhomogeneous Poisson process model that can be used to derive most existing models in the software reliability literature. Section 25.3 describes a generalized random field environment (RFE) model incorporating both the testing phase and operating phase in the software development cycle for estimating the reliability of software systems in the field. In contrast to some existing models that assume the same software failure rate for the software

Many software reliability models have been proposed to help software developers and managers understand and analyze the software development process, estimate the development cost, and assess the level of software reliability. Among these software reliability models, models based on the nonhomogeneous Poisson process (NHPP) have been successfully applied to model the software failure processes that possess certain trends such as reliability growth or deterioration. NHPP models seem to be useful to predict software failures and software reliability in terms of time and to determine when to stop testing and release the software [1].

Currently most existing NHPP software reliability models have been carried out through the fault intensity rate function and the mean-value functions (MVF) $m(t)$ within a controlled operating environment [2–55]. Obviously, different models use different assumptions and therefore provide different mathematical forms for the mean-value function $m(t)$. Table 25.1 shows a summary of several existing models appearing in the software reliability engineering literature [14]. Generally, these models are applied to software testing data and then to make predictions of software failures and reliability

H. Pham (✉)
 Department of Industrial and Systems Engineering, Rutgers University, Piscataway, NJ, USA
 e-mail: hopham@soe.rutgers.edu
 X. Teng
 Dotdash Meredith, New York, NY, USA
 e-mail: xiaolin93@yahoo.com

Table 25.1 Summary of NHPP software reliability models [44, 50]

Model name	MVF $[m(t)]$
Goel–Okumoto (G–O)	$m(t) = a(1 - e^{-bt})$
Delayed S-shaped	$m(t) = a[1 - (1 + bt)e^{-bt}]$
Inflection S-shaped SRGM	$m(t) = \frac{a(1 - e^{-bt})}{1 + \beta e^{-bt}}$
HD/G–O model	$m(t) = \log \left[(e^a - c) / (e^{ae^{-bt}} - c) \right]$
Yamada exponential	$m(t) = a \left(1 - e^{-r\alpha(1 - e^{-(\beta t)})} \right)$
Yamada Rayleigh	$m(t) = a \left(1 - e^{-r\alpha \left(1 - e^{-(\beta t^2/2)} \right)} \right)$
Yamada imperfect debugging model (1)	$m(t) = \frac{ab}{\alpha + b} (e^{\alpha t} - e^{-bt})$
Yamada imperfect debugging model (2)	$m(t) = a \left(1 - e^{-bt} \right) \left(1 - \frac{\alpha}{b} \right) + \alpha at$
PNZ model	$m(t) = \frac{a}{1 + \beta} e^{-bt} \left[(1 - e^{-bt}) \left(1 - \frac{\alpha}{b} \right) + \alpha at \right]$
Pham–Zhang model	$m(t) = \frac{1}{1 + \beta e^{-bt}} \left[(c + a) (1 - e^{-bt}) - \frac{a}{b - \alpha} (e^{-\alpha t} - e^{-bt}) \right]$
Dependent-parameter model (Model 1) [26]	$m(t) = \alpha(1 + \gamma t)(\gamma t + e^{-\gamma t} - 1)$
Dependent-parameter model with $m(t_0) \neq 0$ (Model 1a) [26]	$m(t) = m_0 \left(\frac{\gamma t + 1}{\gamma t_0 + 1} \right) e^{-\gamma(t-t_0)} + \alpha(\gamma t + 1) [\gamma t - 1 + (1 - \gamma t_0) e^{-\gamma(t-t_0)}]$
Vtub-shaped fault-detection rate model [28]	$m(t) = N \left(1 - \left(\frac{\beta}{\beta + a^b - 1} \right)^\alpha \right)$
Pham Inflexion model [29]	$m(t) = N \left(1 - \frac{1}{\left(\frac{\beta + e^{bt}}{1 + \beta} \right)^{\frac{a}{b}}} \right)$
Logistic fault-detection model [44]	$m(t) = \frac{a}{1 + d \left(\frac{1 + \beta}{\beta + e^{bt}} \right)}$
Pham-Zhang model [8]	$m(t) = \frac{1}{1 + \beta e^{-bt}} \left((c + a) (1 - e^{-bt}) - \frac{a}{b - \alpha} (e^{-\alpha t} - e^{-bt}) \right)$

in the field. The underlying assumption for this application is that the field environments are the same as, or close to, a testing environment; this is valid for some software systems that are only used in one environment throughout their entire lifetime. However, this assumption is not valid for many applications where a software program may be used in many different locations once it is released.

The operating environments for the software in the field are quite different. The randomness of the field environment will affect software failure and software reliability in an unpredictable way. *Yang* and *Xie* [15] mentioned that the operational reliability and testing reliability are often different from each other, but they assumed that the operational failure rate is still close to the testing failure rate, and hence that the difference between them is that the operational failure rate decreases with time, while the testing failure rate remains constant. *Zhang* et al. [16] proposed an NHPP software reliability calibration model by introducing a calibration factor. This calibration factor, K , obtained from software failures in both the testing and field operation phases will be a multiplier to the software failure intensity. This calibrated software reliability model can be used to assess and adjust the predictions of software reliability in the operation phase.

Instead of relating the operating environment to the failure intensity λ , in this chapter we assume that the effect of

the operating environment is to multiply the unit failure-detection rate $b(t)$ achieved in the testing environment using the concept of the proportional hazard approach suggested by *Cox* [56]. If the operating environment is more liable to software failure, then the unit fault-detection rate increases by some factor η greater than 1. Similarly, if the operating environment is less liable to software failure, then the unit fault-detection rate decreases by some positive factor η less than 1.

This chapter describes a model based on the NHPP model framework for predicting software failures and evaluating the software reliability in random field environments. A generalized random field environment (RFE) model incorporating both the testing phase and operating phase in the software development cycle with Vtub-shaped fault-detection rate is discussed. An explicit solution of the mean value function for this model is derived. Numerical results of some selected NHPP models are also discussed based on existing criteria such as mean squared error (MSE), predictive power, predictive-ratio risk, and normalized criteria distance from a set of software failure data.

Based on this model, developers and engineers can further develop specific software reliability models customized to various applications.

Notations	
$R(t)$	Software reliability function
η	Random environmental factor
$G(\eta)$	Cumulative distribution function of η
γ	Shape parameter of gamma distributions
θ	Scale parameter of gamma distributions
α, β	Parameters of beta distributions
$N(t)$	Counting process which counts the number of software failures discovered by time t
$m(t)$	Expected number of software failures detected by time t , $m(t) = E[N(t)]$
$a(t)$	Expected number of initial software faults plus introduced faults by time t
$m_1(t)$	Expected number of software failures in testing by time t
$m_2(t)$	Expected number of software failures in the field by time t
$a_1(t)$	Expected number of initial software faults plus introduced faults discovered in the testing by time t
a	Number of initial software faults at the beginning of testing phase, is a <i>software parameter</i> that is directly related to the software itself
T	Time to stop testing and release the software for field operations
a_F	Number of initial software faults in the field (at time T)
$b(t)$	Failure detection rate per fault at time t , is a <i>process parameter</i> that is directly related to testing and failure process
p	Probability that a fault will be successfully removed from the software
q	Error introduction rate at time t in the testing phase
MLE	Maximum likelihood estimation
RFE-model	Software reliability model subject to a random field environment
γ -RFE	Software reliability model with a gamma distributed field environment
β -RFE	Software reliability model with a beta distributed field environment
NHPP	Nonhomogeneous Poisson process
SRGM	Software reliability growth model
HD	Hossain–Ram
PNZ	Pham–Nordman–Zhang
G–O	Goel–Okumoto
MLE	Maximum likelihood estimation
RFE	Random field environment

25.1 A Generalized NHPP Software Reliability Model

A generalized NHPP model studied by Zhang et al. [7] can be formulated as follows:

$$m'(t) = \eta b(t) [a(t) - pm(t)], \tag{25.1}$$

$$a'(t) = q \cdot m'(t), \tag{25.2}$$

where $m(t)$ is the number of software failures expected to be detected by time t . If the marginal conditions are given as $m(0) = 0$ and $a(0) = a$, then for a specific environmental factor η , the solutions to (25.1) and (25.2) are, given in [7], as follows

$$m_\eta(t) = a \int_0^t \eta b(u) e^{-\int_0^u \eta(p-q)b(\tau) d\tau} du, \tag{25.3}$$

$$a_\eta(t) = a \left[1 + \int_0^t \eta q b(u) e^{-\int_0^u \eta(p-q)b(\tau) d\tau} du \right]. \tag{25.4}$$

This is the generalized form of the NHPP software reliability model. When $p = 1$, $\eta = 1$, and $q = 0$, then for any given function $a(t)$ and $b(t)$, all the functions listed in Table 25.1 can easily be obtained.

25.2 Generalized Random Field Environment (RFE) Model

The testing environment is often a controlled environment with much less variation compared to the field environments, which may be quite different for the field application software. Once a software program is released, it may be used in many different locations and various applications in industries. The operating environments for the software are quite different. Therefore, the randomness of the field environment will affect the cumulative software failure data in an unpredictable way.

Figure 25.1 shows the last two phases of the software life cycle: in-house testing and field operation [18]. If T is the time to stop testing and release the software for field operations, then the time period $0 \leq t \leq T$ refers to the time period of *software testing*, while the time period $T \leq t$ refers to the postrelease period – *field operation*.

The environmental factor η is used to capture the uncertainty about the environment and its effects on the software failure rate. In general, software testing is carried out in a controlled environment with very small variations, which can be used as a reference environment where η is constant and equals to 1. For the field operating environment, the

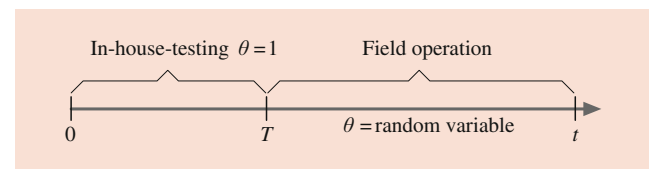


Fig. 25.1 Testing versus field environment where T is the time to stop testing and release the software

environmental factor η is assumed to be a non-negative random variable (RV) with probability density function (PDF) $f(\eta)$, that is,

$$\eta = \begin{cases} 1 & t \leq T \\ \text{RV with PDF } f(\eta) & t \geq T \end{cases} \quad (25.5)$$

If the value of η is less than 1, this indicates that the conditions are less favorable to fault detection than that of testing environment. Likewise, if the value of η is greater than 1, it indicates that the conditions are more favorable to fault detection than that of the testing environment.

From (25.3) and (25.5), the mean-value function and the function $a_1(t)$ during testing can be obtained as

$$\begin{aligned} m_1(t) &= a \int_0^t b(u) e^{-\int_0^u (p-q)b(\tau) d\tau} du \quad t \leq T, \\ a_1(t) &= a \left[1 + \int_0^t qb(u) \times e^{-\int_0^u (p-q)b(\tau) d\tau} du \right] \quad t \leq T. \end{aligned} \quad (25.6)$$

For the field operation where $t \geq T$, the mean-value function can be represented as

$$\begin{aligned} m_2(t) &= m_1(T) + \int_0^\infty m_\eta(t) f(\eta) d\eta = m_1(T) \\ &+ \int_0^\infty \left[a_F \int_T^t \eta b(u) \times e^{-\int_T^u \eta (p-q)b(\tau) d\tau} du \right] f(\eta) d\eta \quad t \geq T \\ &= m_1(T) + \int_T^t a_F b(u) \left[\int_0^\infty \eta \times e^{-\eta \int_T^u (p-q)b(\tau) d\tau} f(\eta) d\eta \right] du, \end{aligned} \quad (25.7)$$

where a_F is the number of faults in the software at time T . Using the Laplace transform formula, the mean-value function can be rewritten as

$$\begin{aligned} m_2(t) &= m_1(T) + \int_T^t a_F b(u) \times \left(-\frac{dF^*(s)}{ds} \Big|_{s=\int_0^u (p-q)b(\tau) d\tau} \right) du, \\ t &\geq T \\ &= m_1(T) + \frac{a_F}{(p-q)} \times \int_T^t \left\{ -dF^* \left[(p-q) \int_T^u b(\tau) d\tau \right] \right\}, \end{aligned}$$

where $F^*(s)$ is the Laplace transform of the PDF $f(x)$ and

$$\int_0^\infty xe^{-x \cdot s} f(x) dx = -\frac{dF^*(s)}{ds}$$

or, equivalently,

$$\begin{aligned} m_2(t) &= m_1(T) - \frac{a_F}{(p-q)} \times F^* \left[(p-q) \int_T^t b(\tau) d\tau \right] \Big|_T^t, \quad t \leq T \\ &= m_1(T) + \frac{a_F}{(p-q)} \times \left\{ F^*(0) - F^* \left[(p-q) \int_T^t b(\tau) d\tau \right] \right\}. \end{aligned}$$

Notice that $F^*(0) = \int_0^\infty e^{-0x} f(x) dx = 1$, so

$$\begin{aligned} m_2(t) &= m_1(T) + \frac{a_F}{(p-q)} \\ &\times \left\{ 1 - F^* \left[(p-q) \int_T^t b(\tau) d\tau \right] \right\} \quad t \geq T. \end{aligned}$$

The expected number of faults in the software at time T is given by

$$\begin{aligned} a_F &= a_1(T) - pm_1(T) \\ &= a \left[1 - \int_0^T (p-q) b(u) e^{-\int_0^u (p-q)b(\tau) d\tau} du \right] \\ &= a e^{-\int_0^T (p-q)b(\tau) d\tau}. \end{aligned}$$

The generalized RFE model can be obtained as

$$m(t) = \begin{cases} \frac{a}{(p-q)} \left(1 - e^{-(p-q) \int_0^t b(\tau) d\tau} \right) & t \leq T \\ \frac{a}{(p-q)} \left\{ 1 - e^{-(p-q) \int_0^T b(\tau) d\tau} \times F^* \left[(p-q) \int_T^t b(\tau) d\tau \right] \right\} & t \geq T. \end{cases} \quad (25.8)$$

The model in (25.8) is a generalized software reliability model subject to random field environments. The next section presents specific RFE models for the gamma and beta distributions of the random field environmental factor η .

25.3 RFE Software Reliability Models

Obviously, the environmental factor η must be non-negative. Any suitable non-negative distribution may be used to describe the uncertainty about η . In this section, we present two RFE models. The first model is a γ -RFE model, based on the gamma distribution, which can be used to evaluate and predict software reliability in field environments where the software failure-detection rate can be either greater or less than the failure detection rate in the testing environment. The second model is a β -RFE model, based on the beta distribution, which can be used to predict software reliability in field environments where the software failure detection rate can only be less than the failure detection rate in the testing environment.

25.3.1 γ -RFE Model

In this model, we use the gamma distribution to describe the random environmental factor η . This model is called the γ -RFE model.

Assume that η follows a gamma distribution with a probability density function as follows:

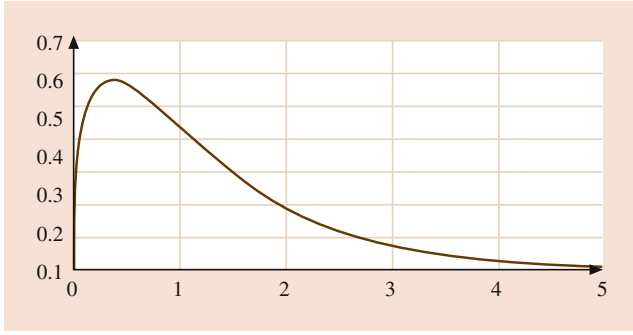


Fig. 25.2 A gamma density function

$$f_{\gamma}(\eta) = \frac{\theta^{\gamma} \eta^{\gamma-1} e^{-\theta \cdot \eta}}{\Gamma(\gamma)}, \quad \gamma, \theta > 0; \eta \geq 0. \quad (25.9)$$

The gamma distribution has sufficient flexibility and has desirable qualities with respect to computations [18]. Figure 25.2 shows an example of the gamma density probability function. The gamma function seems to be reasonable to describe a software failure process in those field environments where the software failure-detection rate can be either greater (i.e., $\eta > 1$) or less than (i.e., $\eta < 1$) the failure-detection rate in the testing environment.

The Laplace transform of the probability density function in (25.9) is

$$F^*(s) = \left(\frac{\theta}{\theta + s} \right)^{\gamma}. \quad (25.10)$$

Assume that the error-detection rate function $b(t)$ is given by

$$b(t) = \frac{b}{1 + ce^{-bt}}. \quad (25.11)$$

where b is the asymptotic unit software-failure detection rate and c is the parameter defining the shape of the learning curve, then from (25.8) the mean-value function of the γ -RFE model can be obtained as follows

$$m_{\gamma}(t) = \begin{cases} \frac{a}{(p-q)} \left[1 - \left(\frac{1+c}{e^{bt}+c} \right)^{(p-q)} \right] & t \leq T, \\ \frac{a}{(p-q)} \left[1 - \left(\frac{1+c}{e^{bt}+c} \right)^{(p-q)} \times \left(\frac{\theta}{\theta + (p-q) \ln \left(\frac{c+e^{bt}}{c+e^{bT}} \right)} \right)^{\gamma} \right] & t \geq T. \end{cases} \quad (25.12)$$

25.3.2 β -RFE Model

This section presents a model using the beta distribution that describes the random environmental factor η , called the β -RFE model.

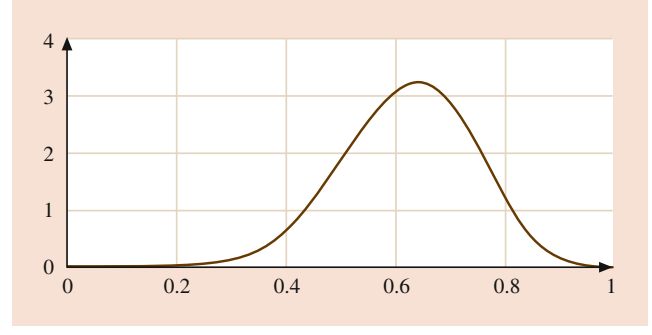


Fig. 25.3 A PDF curve of the beta distribution

The beta PDF is

$$f_{\beta}(\eta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha) \Gamma(\beta)} \eta^{\alpha-1} (1 - \eta)^{\beta-1}, \quad \alpha > 0, \beta > 0; 0 \leq \eta \leq 1. \quad (25.13)$$

Figure 25.3 shows an example of the beta density function. It seems that the β -RFE model is a reasonable function to describe a software failure process in those field environments where the software failure-detection rate can only be less than the failure-detection rate in the testing environment. This is not uncommon in the software industry because, during software testing, the engineers generally test the software intensely and conduct an *accelerated* test on the software in order to detect most of the software faults as early as possible.

The Laplace transform of the PDF in (25.13) is

$$F_{\beta}^*(s) = e^{-s} \cdot \text{HG}([\beta], [\alpha + \beta], s), \quad (25.14)$$

where $\text{HG}([\beta], [\alpha + \beta], s)$ is a generalized hypergeometric function such that

$$\text{HG}([a_1, a_2, \dots, a_m], [b_1, b_2, \dots, b_n], s) = \sum_{k=0}^{\infty} \left(\frac{s^k \prod_{i=1}^m \frac{\Gamma(a_i+k)}{\Gamma(a_i)}}{\prod_{i=1}^n \frac{\Gamma(b_i+k)}{\Gamma(b_i)}} \right).$$

Therefore,

$$\begin{aligned} F_{\beta}^*(s) &= e^{-s} \sum_{k=0}^{\infty} \left(\frac{\Gamma(\alpha + \beta) \Gamma(\beta + k) s^k}{\Gamma(\beta) \Gamma(\alpha + \beta + k) k!} \right) \\ &= \sum_{k=0}^{\infty} \left(\frac{\Gamma(\alpha + \beta) \Gamma(\beta + k) s^k e^{-s}}{\Gamma(\beta) \Gamma(\alpha + \beta + k) k!} \right) \\ &= \sum_{k=0}^{\infty} \left(\frac{\Gamma(\alpha + \beta) \Gamma(\beta + k)}{\Gamma(\beta) \Gamma(\alpha + \beta + k)} \text{Poisson}(k, s) \right). \end{aligned}$$

where the Poisson PDF is given by

$$\text{Poisson}(k, s) = \frac{s^k e^{-s}}{k!}.$$

Using the same error-detection rate function in (25.11) and replacing $F^*(s)$ by $F_{\beta}^*(s)$, the mean-value function of the β -RFE model is

$$m_{\beta}(t) = \begin{cases} \frac{a}{(p-q)} \left[1 - \left(\frac{1+c}{e^{bt}+c} \right)^{(p-q)} \right] & t \leq T, \\ \frac{a}{(p-q)} \left[1 - \left(\frac{1+c}{e^{bT}+c} \right)^{(p-q)} \right] \times \sum_{k=0}^{\infty} \left(\frac{\Gamma(\alpha+\beta)\Gamma(\beta+k)\text{Poisson}(k,s)}{\Gamma(\beta)\Gamma(\alpha+\beta+k)} \right) & t \geq T. \end{cases} \quad (25.15)$$

where

$$s = (p - q) \left[\ln \left(\frac{c + e^{bt}}{c + e^{bT}} \right) \right].$$

The next section will discuss the parameter estimation and illustrate the applications of these two RFE software reliability models using software failure data.

25.4 Parameter Estimation

25.4.1 Maximum Likelihood Estimation (MLE)

We use the MLE method to estimate the parameters in these two RFE models. Let y_i be the cumulative number of software faults detected up to time t_i , $i = 1, 2, \dots, n$. Based on the NHPP, the likelihood function is given by

$$L = \prod_{i=1}^n \frac{[m(t_i) - m(t_{i-1})]^{y_i - y_{i-1}}}{(y_i - y_{i-1})!} e^{-[m(t_i) - m(t_{i-1})]}. \quad (25.16)$$

The logarithmic form of the above likelihood function is

$$\begin{aligned} \ln L &= \sum_{i=1}^n \{y_i - y_{i-1}\} \ln [m(t_i) - m(t_{i-1})] \\ &\quad - [m(t_i) - m(t_{i-1})] - \ln [(y_i - y_{i-1})!]. \end{aligned} \quad (25.17)$$

In this analysis, the error-removal efficiency p is given. Each model has five unknown parameters. For example, in the γ -RFE model, we need to estimate the following five unknown parameters: a, b, q, γ , and θ . For the β -RFE model, we need to estimate: a, b, q, α , and β . By taking derivatives of (25.17) with respect to each parameter and setting the results equal to zero, we can obtain five equations for each RFE model. After solving all those equations, we obtain the maximum likelihood estimates (MLEs) of all parameters for each RFE model.

Table 25.2 shows a set of failure data from a telecommunication software application during software testing [16]. The column “Time” shows the normalized cumulative time spent in software testing for this telecommunication application,

Table 25.2 Normalized cumulative failures and times during software testing

Time	Failures	Time	Failures	Time	Failures
0.0001	0.0249	0.0038	0.3483	0.0121	0.6766
0.0002	0.0299	0.0044	0.3532	0.0128	0.7015
0.0002	0.0647	0.0048	0.3682	0.0135	0.7363
0.0003	0.0647	0.0053	0.3881	0.0142	0.7761
0.0005	0.1095	0.0058	0.4478	0.0147	0.7761
0.0006	0.1194	0.0064	0.4876	0.0155	0.8159
0.0008	0.1443	0.0070	0.5224	0.0164	0.8259
0.0012	0.1692	0.0077	0.5473	0.0172	0.8408
0.0016	0.1990	0.0086	0.5821	0.0176	0.8458
0.0023	0.2289	0.0095	0.6119	0.0180	0.8756
0.0028	0.2637	0.0105	0.6368	0.0184	0.8955
0.0033	0.3134	0.0114	0.6468	0.0184	0.9005

Table 25.3 Normalized cumulative failures and their times in operation

Time	Failures	Time	Failures	Time	Failures
0.0431	0.9055	0.3157	0.9751	0.7519	0.9900
0.0616	0.9104	0.3407	0.9751	0.7585	0.9900
0.0801	0.9204	0.3469	0.9751	0.7718	0.9900
0.0863	0.9254	0.3967	0.9751	0.7983	0.9900
0.1357	0.9303	0.4030	0.9801	0.8251	0.9900
0.1419	0.9353	0.4291	0.9851	0.8453	0.9900
0.1666	0.9453	0.4357	0.9851	0.8520	0.9900
0.2098	0.9453	0.4749	0.9851	0.9058	0.9900
0.2223	0.9502	0.5011	0.9851	0.9126	0.9900
0.2534	0.9502	0.5338	0.9851	0.9193	0.9900
0.2597	0.9502	0.5731	0.9851	0.9395	0.9950
0.2659	0.9502	0.6258	0.9900	0.9462	0.9950
0.2721	0.9552	0.6656	0.9900	0.9529	1.0000
0.2971	0.9602	0.6789	0.9900	0.9865	1.0000
0.3033	0.9701	0.7253	0.9900	1.0000	1.0000

and the column “Failures” shows the normalized cumulative number of failures occurring in the testing period up to the given time.

The time to stop testing is $T = 0.0184$. After the time T , the software is released for field operations. Table 25.3 shows the field data for this software release. Similarly, the column “Time” shows the normalized cumulative time spent in the field for this software application, and the time in Table 25.3 is continued from the time to stop testing T . The column “Failures” shows the normalized cumulative number of failures found after releasing the software for field operations up to the given time. The cumulative number of failures is the total number of software failures since the beginning of software testing.

To obtain a better understanding of the software development process, we show the actual results of the MLE solutions instead of the normalized results. In this study, let us assume that testing engineers have a number of years of

Table 25.4 MLE solutions for the γ -RFE model

\hat{a}	\hat{b}	\hat{q}	\hat{c}	$\hat{\gamma}$	$\hat{\theta}$
236.58	0.001443	0	0	0.2137	10.713

Table 25.5 MLE solutions for the β -RFE model

\hat{a}	\hat{b}	\hat{q}	\hat{c}	$\hat{\alpha}$	$\hat{\beta}$
236.07	0.001449	0	0	0.1862	8.6922

experience of this particular product and software development skills and therefore conducted perfect debugging during the test. In other word, $p = 1$. The maximum likelihood estimates of all the parameters in the γ -RFE model are obtained as shown in Table 25.4.

Similarly, set $p = 1$, the MLE of all the parameters in the β -RFE model are obtained as shown in Table 25.5.

For both RFE models, the MLE results can be used to obtain more insightful information about the software development process. In this example, at the time to stop testing the software $T = 0.0184$, the estimated number of remaining faults in the system is $a_F = a - (p - q)m(T) = 55$.

25.4.2 Mean-Value Function Fits

After we obtain the MLEs for all the parameters, we can plot the mean-value function curve fits for both the γ -RFE and β -RFE models based on the MLE parameters against the actual software application failures.

Table 25.6 shows the mean-value function curve fits for both the models where the columns $m_\gamma(t)$ and $m_\beta(t)$ show the mean-value function for the γ -RFE model and the β -RFE model, respectively.

The γ -RFE and β -RFE models yield very close fits and predictions on software failures. Figure 25.4 shows the mean-value function curve fits for both the γ -RFE model and β -RFE model. Both models appear to be a good fit for the given data set. Since we are particularly interested in the fits and the predictions for software failure data during field operation, we also plot the detailed mean-value curve fits for both the γ -RFE model and the β -RFE model in Fig. 25.5.

For the overall fitting of the mean-value function against the actual software failures, the MSE is 23.63 for the γ -RFE model fit and is 23.69 for the β -RFE model. We can also obtain the fits and predictions for software failures by applying some existing NHPP software reliability models to the same set of failure data. Since all these existing models assume a constant failure-detection rate throughout both the software testing and operation periods, we only apply the software testing data to the software models and then predict the software failures in the field environments.

Table 25.6 The mean-value functions for both RFEs models

Time	Failures	$m_\gamma(t)$	$m_\beta(t)$	Time	Failures	$m_\gamma(t)$	$m_\beta(t)$
0.0000	0.0000	0.0000	0.0000	0.1357	0.9303	0.9340	0.9341
0.0001	0.0249	0.0085	0.0085	0.1419	0.9353	0.9352	0.9354
0.0002	0.0299	0.0152	0.0152	0.1666	0.9453	0.9398	0.9399
0.0002	0.0647	0.0219	0.0219	0.2098	0.9453	0.9469	0.9467
0.0003	0.0647	0.0302	0.0302	0.2223	0.9502	0.9487	0.9485
0.0005	0.1095	0.0466	0.0467	0.2534	0.9502	0.9530	0.9525
0.0006	0.1194	0.0547	0.0548	0.2597	0.9502	0.9538	0.9533
0.0008	0.1443	0.0708	0.0709	0.2659	0.9502	0.9545	0.9540
0.0012	0.1692	0.1023	0.1025	0.2721	0.9552	0.9553	0.9547
0.0016	0.1990	0.1404	0.1406	0.2971	0.9602	0.9582	0.9575
0.0023	0.2289	0.1915	0.1917	0.3033	0.9701	0.9589	0.9582
0.0028	0.2637	0.2332	0.2335	0.3157	0.9751	0.9603	0.9594
0.0033	0.3134	0.2667	0.2670	0.3407	0.9751	0.9628	0.9618
0.0038	0.3483	0.3053	0.3056	0.3469	0.9751	0.9635	0.9624
0.0044	0.3532	0.3422	0.3426	0.3967	0.9751	0.9681	0.9667
0.0048	0.3682	0.3718	0.3721	0.4030	0.9801	0.9686	0.9672
0.0053	0.3881	0.4003	0.4007	0.4291	0.9851	0.9708	0.9692
0.0058	0.4478	0.4332	0.4336	0.4357	0.9851	0.9713	0.9697
0.0064	0.4876	0.4648	0.4651	0.4749	0.9851	0.9743	0.9725
0.0070	0.5224	0.4998	0.5002	0.5011	0.9851	0.9761	0.9742
0.0077	0.5473	0.5332	0.5335	0.5338	0.9851	0.9783	0.9762
0.0086	0.5821	0.5772	0.5775	0.5731	0.9851	0.9808	0.9785
0.0095	0.6119	0.6205	0.6208	0.6258	0.9900	0.9839	0.9813
0.0105	0.6368	0.6600	0.6602	0.6656	0.9900	0.9860	0.9833
0.0114	0.6468	0.6953	0.6955	0.6789	0.9900	0.9867	0.9839
0.0121	0.6766	0.7210	0.7211	0.7253	0.9900	0.9890	0.9860
0.0128	0.7015	0.7479	0.7479	0.7519	0.9900	0.9902	0.9871
0.0135	0.7363	0.7684	0.7684	0.7585	0.9900	0.9905	0.9874
0.0142	0.7761	0.7924	0.7924	0.7718	0.9900	0.9911	0.9879
0.0147	0.7761	0.8050	0.8049	0.7983	0.9900	0.9923	0.9890
0.0155	0.8159	0.8294	0.8292	0.8251	0.9900	0.9934	0.9900
0.0164	0.8259	0.8522	0.8520	0.8453	0.9900	0.9943	0.9908
0.0172	0.8408	0.8713	0.8710	0.8520	0.9900	0.9945	0.9910
0.0176	0.8458	0.8804	0.8801	0.9058	0.9900	0.9966	0.9929
0.0180	0.8756	0.8897	0.8893	0.9126	0.9900	0.9969	0.9932
0.0184	0.8955	0.8987	0.8983	0.9193	0.9900	0.9971	0.9934
0.0184	0.9005	0.8995	0.8991	0.9395	0.9950	0.9979	0.9941
0.0431	0.9055	0.9092	0.9092	0.9462	0.9950	0.9981	0.9943
0.0616	0.9104	0.9153	0.9155	0.9529	1.0000	0.9983	0.9945
0.0801	0.9204	0.9208	0.9210	0.9865	1.0000	0.9995	0.9956
0.0863	0.9254	0.9224	0.9227	1.0000	1.0000	1.0000	0.9960

Figure 25.6 shows the comparisons of the mean-value function curve fits between the two RFE models and some existing NHPP software reliability models. It appears that the two models that include consideration of the field environment on the software failure-detection rate perform better in terms of the predictions for software failures in the field.

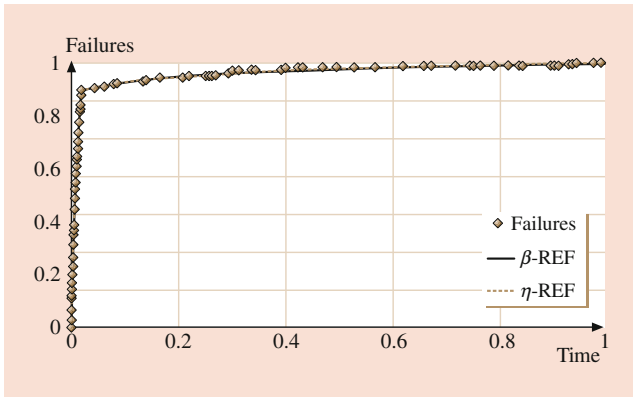


Fig. 25.4 Mean-value function curve fits for both RFE models

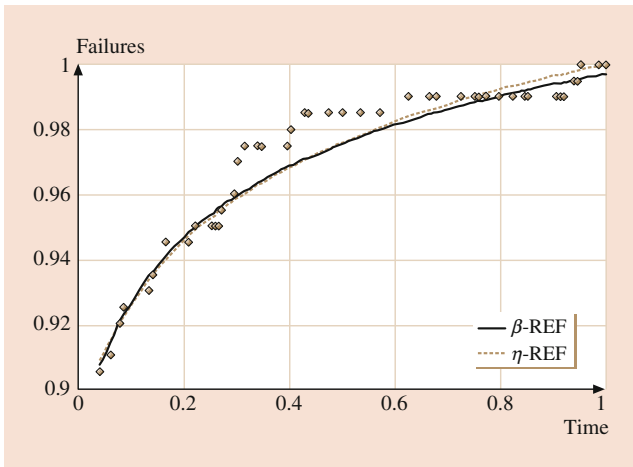


Fig. 25.5 Mean-value function fitting comparisons

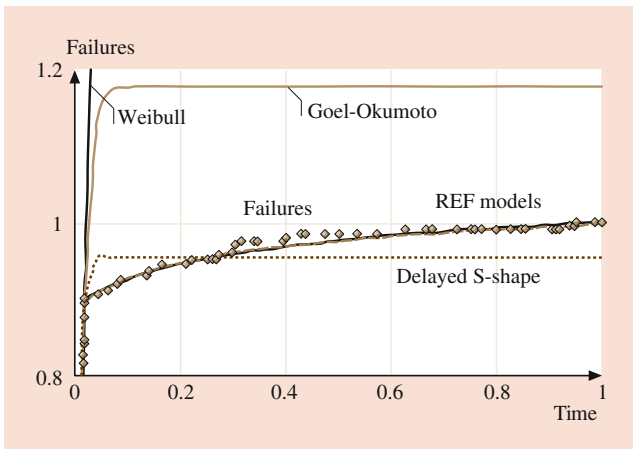


Fig. 25.6 Model comparisons

25.4.3 Software Reliability

Once the MLEs of all the parameters in (25.12) and (25.14) are obtained, the software reliability within $(t, t + x)$ can be determined as

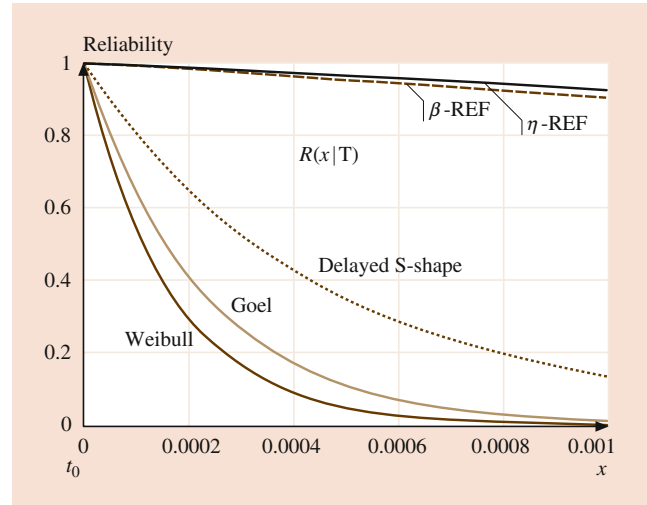


Fig. 25.7 Reliability prediction comparisons

$$R(x|t) = e^{-[m(t+x)-m(t)]}. \tag{25.18}$$

Let $T = 0.0184$, and change x from 0 to 0.004, then we can compare the reliability predictions between the two RFE models and some other NHPP models that assume a constant failure-detection rate for both software testing and operation. The reliability prediction curves are shown in Fig. 25.7. From Fig. 25.7, we can see that the NHPP models without consideration of the environmental factor yield much lower predictions for software reliability in the field than the two proposed RFE software reliability models.

25.4.4 Confidence Interval

γ -RFE Model

To see how good the reliability predictions given by the two RFE models are, in this section we describe how to construct confidence intervals for the prediction of software reliability in the random field environments. From Tables 25.4 and 25.5, the MLEs of c and q are equal to zero and, if p is set to 1, then the model in (25.12) becomes

$$m(t) = \begin{cases} a(1 - e^{-b \cdot t}) & t \leq T, \\ a \left[1 - e^{-b \cdot t} \left(\frac{\theta}{\theta + b(t-T)} \right)^\gamma \right] & t \geq T. \end{cases} \tag{25.19}$$

This model leads to the same MLE results for the parameters a, b, γ , and θ and also yields exactly the same mean-value function fits and predictions as the model in (25.12). To obtain the confidence interval for the reliability predictions for the γ -RFE model, we derive the variance-covariance matrix for all the maximum likelihood estimates as follows.

If we use $x_i, i = 1, 2, 3,$ and $4,$ to denote all the parameters in the model, or

$$x_1 \rightarrow a \quad x_2 \rightarrow b \quad x_3 \rightarrow \theta \quad x_4 \rightarrow \gamma.$$

The Fisher information matrix H can be obtained as

$$H = \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{pmatrix},$$

where

$$h_{ij} = E \left(-\frac{\partial^2 L}{\partial x_i \partial x_j} \right) i, j = 1, \dots, 6.$$

where L is the log-likelihood function in (25.18).

If we denote $z(t_k) = m(t_k) - m(t_{k-1})$ and $\Delta y_k = y_k - y_{k-1}, k = 1, 2, \dots, n,$ then we have

$$\begin{aligned} \frac{\partial^2 L}{\partial x_i \partial x_j} &= \sum_{k=1}^n \left[-\frac{\Delta y_k}{z(t_k)^2} \frac{\partial z(t_k)}{\partial x_i} \cdot \frac{\partial z(t_k)}{\partial x_j} \right. \\ &\quad \left. + \left(\frac{\Delta y_k - z(t_k)}{z(t_k)} \cdot \frac{\partial^2 z(t_k)}{\partial x_i \partial x_j} \right) \right]. \end{aligned}$$

Then we can obtain each element in the Fisher information matrix $H.$ For example,

$$\begin{aligned} h_{11} &= E \left(-\frac{\partial^2 L}{\partial a^2} \right) \\ &= \sum_{k=1}^n \left\{ \sum_{\Delta y_k=0}^{\infty} \left[\frac{\Delta y_k}{z(t_k)^2} \left(\frac{\partial z(t_k)}{\partial a} \right)^2 \right] \times \frac{[z(t_k)]^{\Delta y_k} e^{-z(t_k)}}{(\Delta y_k)!} \right\} \\ &= \sum_{k=1}^n \left\{ \sum_{\Delta y_k=0}^{\infty} \left[\frac{\Delta y_k}{z(t_k)^2} \left(\frac{z(t_k)}{a} \right)^2 \right] \times \frac{[z(t_k)]^{\Delta y_k} e^{-z(t_k)}}{(\Delta y_k)!} \right\} \\ &= \sum_{k=1}^n \left(\frac{1}{a^2} \sum_{\Delta y_k=0}^{\infty} \Delta y_k \frac{[z(t_k)]^{\Delta y_k} e^{-z(t_k)}}{(\Delta y_k)!} \right) \\ &= \sum_{k=1}^n \left[\frac{1}{a^2} \cdot z(t_k) \right] \\ &= \frac{1}{a^2} m(t_n). \end{aligned}$$

The variance matrix, $V,$ can also be obtained

$$V = (H)^{-1} = \begin{pmatrix} v_{11} & v_{12} & v_{13} & v_{14} \\ v_{21} & v_{22} & v_{23} & v_{24} \\ v_{31} & v_{32} & v_{33} & v_{34} \\ v_{41} & v_{42} & v_{43} & v_{44} \end{pmatrix}.$$

The variances of all the estimate parameters are given by

$$\begin{aligned} \text{Var}(\hat{a}) &= \text{Var}(x_1) = v_{11}, \\ \text{Var}(\hat{b}) &= \text{Var}(x_2) = v_{22}, \\ \text{Var}(\hat{\gamma}) &= \text{Var}(x_3) = v_{33}, \\ \text{Var}(\hat{\theta}) &= \text{Var}(x_4) = v_{44}. \end{aligned}$$

The actual numerical results for the γ -RFE model variance matrix are

$$V\gamma = \begin{pmatrix} 703.8472 & -0.005387 & -88.6906 & -2.6861 \\ -0.005387 & 7.3655 \times 10^{-8} & 1.11 \times 10^{-3} & 3.097 \times 10^{-5} \\ -88.6906 & 1.11 \times 10^{-3} & 92.4287 & 1.1843 \\ -2.6861 & 3.097 \times 10^{-5} & 1.1843 & 0.0238 \end{pmatrix}.$$

β -RFE Model

The model in (25.14) can also be simplified given that the estimates of both q and c are equal to zero and p is set to 1. The mean-value function becomes

$$m_{\beta}(t) = \begin{cases} a(1 - e^{-bt}) & t \leq T, \\ a \left[1 - e^{-bt} \times \sum_{k=0}^{\infty} \left(\frac{\Gamma(\alpha+\beta)\Gamma(\beta+k)\text{Poisson}[k,b(t-T)]}{\Gamma(\beta)\Gamma(\alpha+\beta+k)} \right) \right] & t \geq T. \end{cases}$$

This model leads to the same MLE results for the parameters $a, b, \alpha,$ and β and also yields exactly the same mean-value function fits and predictions. To obtain the confidence interval for the reliability predictions for the β -RFE model, we need to obtain the variance–covariance matrix for all the maximum likelihood estimates.

If we use $x_i, i = 1, 2, 3,$ and $4,$ to denote all the parameters in the model, or

$$x_1 \rightarrow a \quad x_2 \rightarrow b \quad x_3 \rightarrow \alpha \quad x_4 \rightarrow \beta,$$

and go through similar steps as for the γ -RFE model, the actual numerical results for the β -RFE model variance matrix can be obtained as

$$V_{\beta} = \begin{pmatrix} 691.2 & -0.00536 & -2.728 & -66.2172 \\ -0.00536 & 7.4485 \times 10^{-8} & 2.671 \times 10^{-5} & 0.00085 \\ -2.7652 & 2.671 \times 10^{-5} & 0.01820 & 0.8295 \\ -66.2172 & 0.00085 & 0.8295 & 60.5985 \end{pmatrix}$$

Confidence Interval of the Reliability Predictions

If we define a partial derivative vector for the reliability $R(x | t)$ in (25.18) as

$$vR(x|t) = \left(\frac{\partial R(x|t)}{\partial x_1}, \frac{\partial R(x|t)}{\partial x_2}, \frac{\partial R(x|t)}{\partial x_3}, \frac{\partial R(x|t)}{\partial x_4} \right)$$

then the variance of $R(x | t)$ in (25.18) can be obtained as

$$\text{Var} [R(x|t)] = vR(x|t) V[vR(x|t)]^T.$$

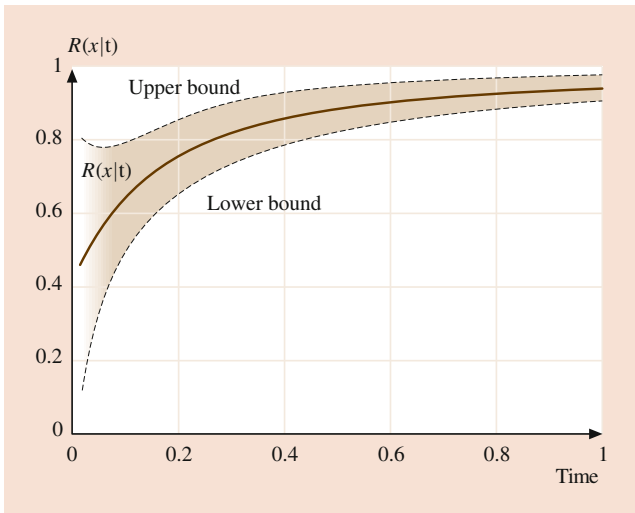


Fig. 25.8 γ -RFE model reliability growth curve and its 95% confidence interval

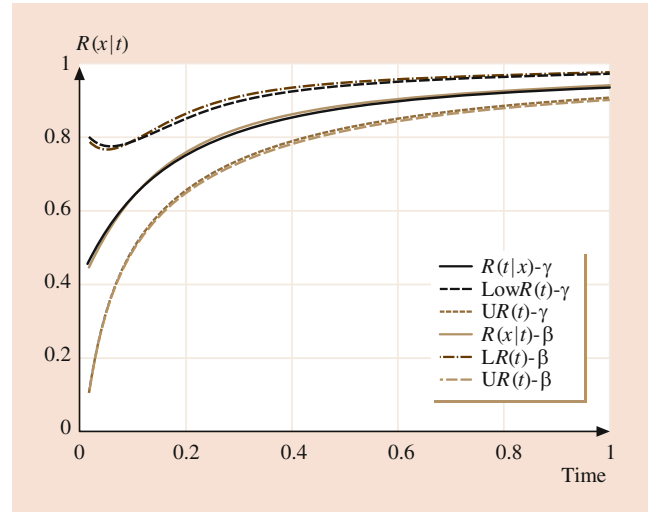


Fig. 25.10 Reliability growth prediction curves and their 95% confidence intervals for the γ -RFE model and the β -RFE model

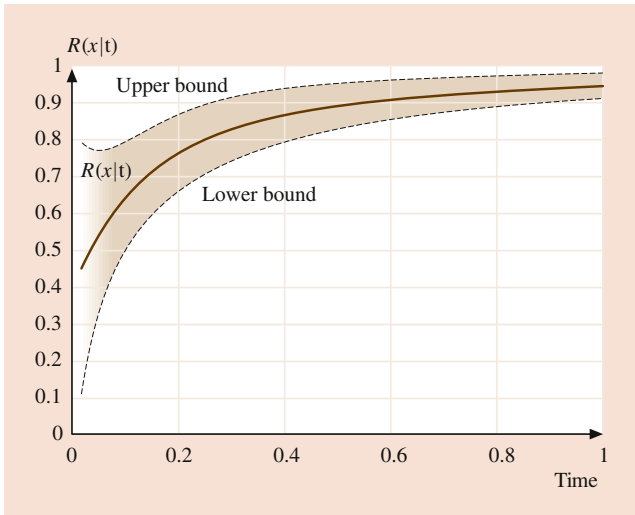


Fig. 25.9 β -RFE model reliability growth prediction and its 95% confidence interval

Assume that the reliability estimation follows a normal distribution, then the 95% confidence interval for the reliability prediction $R(x | t)$ is

$$\left[R(x|t) - 1.96 \times \sqrt{\text{Var} [R(x|t)]}, R(x|t) + 1.96 \times \sqrt{\text{Var} [R(x|t)]} \right]$$

Figures 25.8 and 25.9 show the 95% confidence interval of the reliability predicted by the γ -RFE and β -RFE models, respectively.

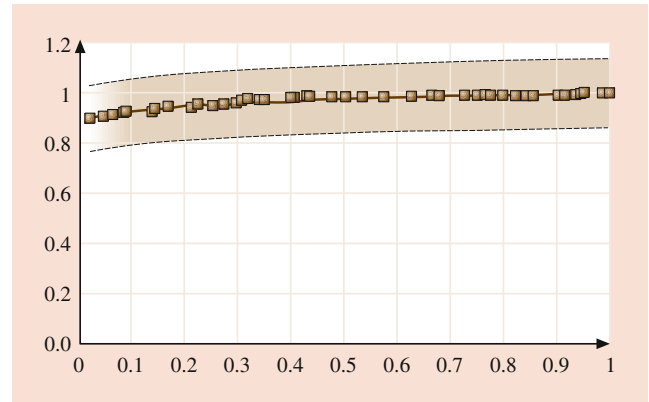


Fig. 25.11 Mean-value function curve fit and its 95% confidence intervals for the γ -RFE model

We plot the reliability predictions and their 95% confidence interval for both the γ -RFE model and the β -RFE model in Fig. 25.10. For this given application data set, the reliability predictions for the γ -RFE model and the β -RFE model are very close to each other, as are their confidence intervals. Therefore, it would not matter too much which one of the two RFE models was used to evaluate the software reliability for this application. However, will these two RFE models always yield similar reliability predictions for all software applications? or which model should one choose for applications if they are not always that close to each other? We will try to answer these two questions in the next section. Figure 25.11 shows the 95% confidence interval for the mean-value function fits and predictions from the γ -RFE model.

Table 25.7 MLEs and fitness comparisons

Parameter	γ -RFE	β -RFE
\hat{a}	236.57930161	236.0745369
\hat{b}	0.001443362	0.001448854
$\hat{\theta}$	10.7160153	
$\hat{\gamma}$	0.213762945	
$\hat{\alpha}$		0.186224489
$\hat{\beta}$		8.692191792
Mean	0.019948	0.020975
Variance	0.0018615	0.002079
MSE	23.63	23.69
Likelihood	-136.1039497	-129.7811199

25.4.5 Concluding and Remarks

Table 25.7 shows all the maximum likelihood estimates of all the parameters and other fitness measures. The maximum likelihood estimates (MLEs) on common parameters, such as a – the initial number of faults in the software and b – the unit software failure-detection rate during testing, are consistent for both models. Both models provide very close predictions for software reliability and also give similar results for the mean and variance of the random environment factor η .

The underlying rationale for this phenomenon is the similarity between the gamma and beta distributions when the random variable η is close to zero. In this application, the field environments are much less liable to software failure than the testing environment. The random field environmental factor, η , is mostly much less than 1 with mean $(\eta) \approx 0.02$.

Figure 25.12 shows the PDF curves of the environmental factor η based on the MLEs of all the parameters for both the γ -RFE model and the β -RFE model. We observe that the PDF curves for the beta and gamma distributions are also very close to each other. The two RFEs models give similar results because this software application is much less likely to fail in the field environment, with mean $(\eta) = 0.02$. If the mean (η) is not so close to 0, then we would expect to have different prediction results from the γ -RFE model and the β -RFE model.

We suggest the following criteria as ways to select between the two models discussed in this chapter for predicting the software reliability in the random field environments:

1. Software less liable to failure in the field than in testing, that is, $\eta \leq 1$

In the γ -RFE model, the random field environmental factor, η following a gamma distribution, ranges from 0 to $+\infty$. For the β -RFE model, the random field environmental factor, η following a beta distribution, ranging from 0 to 1. Therefore, the β -RFE model will be more appropriate for describing field environments in which the software application is likely to fail than in the controlled testing environment.

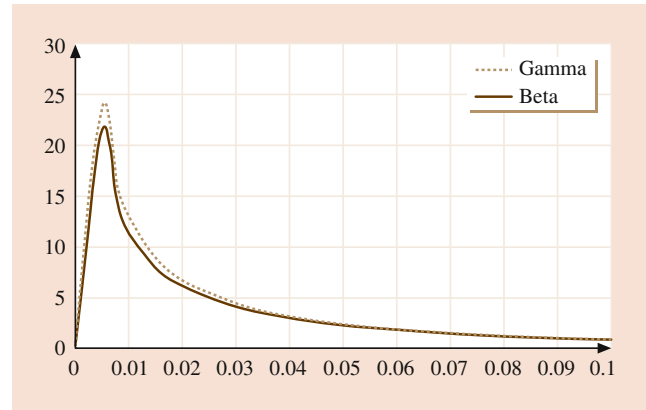


Fig. 25.12 PDF curves comparison for the environmental factor η

For this given application, we notice that when the field environmental factor η is much less than 1 [mean(η) = 0.02], the γ -RFE model yields similar results to the β -RFE model. However, we also observe that the γ -RFE model does not always yield similar results to the β -RFE model when η is not close to 0. In this case, if we keep using the γ -RFE model instead of the β -RFE model, we would expect to see a large variance in the maximum likelihood estimates for all the unknown parameters, and hence a wider confidence interval for the reliability prediction.

2. Smaller variance of the RFE factor η

A smaller variance of the random environmental factor η will generally lead to a smaller confidence interval for the software reliability prediction. It therefore represents a better prediction in the random field environments.
3. Smaller variances for the common parameters a and b

The software parameter a and the process parameter b are directly related to the accuracy of reliability prediction. They can also be used to investigate the software development process. Smaller variances of a and b would lead, in general, to smaller confidence intervals for the mean-value function predictions and reliability predictions.
4. Smaller MSE of the mean-value function fits

A smaller MSE for the mean-value function fits means a better fit of the model to the real system failures. This smaller MSE will usually lead to a better prediction of software failures in random field environments.

The above criteria can be used with care to determine which RFE model should be chosen in practice. They may sometime provide contradictory results. In the case of contradictions, practitioners can often consider selecting the model with the smaller confidence interval for the reliability prediction.

25.5 A RFE Model with Vtub-Shaped Fault-Detection Rate

In this section, we present a specific RFE model with Vtub-shaped fault-detection rate. Numerical results of some selected NHPP models based on MSE, predictive power, predictive-ratio risk, and normalized criteria distance from a set of software failure data are discussed.

Here we assume that a detected fault will be 100% successfully removed from the software during the software testing period and the software fault-detection rate per unit time, $h(t)$, with a Vtub-shaped function [22], is as follows:

$$h(t) = b \ln(a)t^{b-1}a^{t^b} \quad \text{for } a > 1, b > 0 \quad (25.20)$$

We also assume that the random variable η as defined in (25.1) has a gamma distribution with parameters α and β , that is, $\eta \sim \text{gamma}(\alpha, \beta)$ where the PDF of η is given by

$$g(x) = \frac{\beta^\alpha x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)} \quad \text{for } \alpha, \beta > 0; x \geq 0 \quad (25.21)$$

From Eq. (25.1), we can obtain the expected number of software failures detected by time t subject to the uncertainty of the environments as follows:

$$m(t) = N \left(1 - \left(\frac{\beta}{\beta + a^{t^b} - 1} \right)^\alpha \right) \quad (25.22)$$

where N is the expected number of faults that exists in the software before testing.

25.5.1 Model Criteria

We briefly discuss some common criteria such as MSE, predictive-ratio risk (PRR), and predictive power (PP) that will be used to compare the performance of some selected models from Table 25.1 to illustrate the modeling analysis.

The MSE measures the difference between the estimated values and the actual observation and is defined as:

$$\text{MSE} = \frac{\sum_{i=1}^n (\hat{m}(t_i) - y_i)^2}{n - k} \quad (25.23)$$

where y_i = total number of actual failures at time t_i ; $\hat{m}(t_i)$ = the estimated cumulative number of failures at time t_i for $i = 1, 2, \dots, n$; and n and k = number of observations and number of model parameters, respectively.

The predictive-ratio risk (PRR) measures the distance of model estimates from the actual data against the model estimate and is defined as [22]:

$$\text{PRR} = \sum_{i=1}^n \left(\frac{\hat{m}(t_i) - y_i}{\hat{m}(t_i)} \right)^2 \quad (25.24)$$

The predictive power (PP) measures the distance of model estimates from the actual data against the actual data [22]:

$$\text{PP} = \sum_{i=1}^n \left(\frac{\hat{m}(t_i) - y_i}{y_i} \right)^2 \quad (25.25)$$

For all these three criteria – MSE, PRR, and PP – the smaller the value, the better the model fits.

Pham [28] discussed a *normalized criteria distance*, or NCD criteria, to determine the best model from a set of performance criteria. The NCD criteria is defined as follows:

$$D_i = \sum_{j=1}^d \left\{ \left(\left[\sqrt{\sum_{i=1}^2 \left(\frac{C_{ijk}}{\sum_{i=1}^s C_{ijk}} \right)^2} \right] \right) w_j \right\} \quad (25.26)$$

where s and d are the total number of models and criteria, respectively.

w_j = the weight of the j th criteria for $j = 1, 2, \dots, d$

$k = \begin{cases} 1 & \text{represent criteria } j \text{ value} \\ 2 & \text{represent criteria } j \text{ ranking} \end{cases}$

C_{ijl} = the ranking based on specified criterion of model i with respect to (w.r.t.) criteria j

C_{ij2} = criteria value of model i w.r.t. criteria j where $i = 1, 2, \dots, s$ and $j = 1, 2, \dots, d$

Obviously the smaller the NCD value, D_i , it represents the better rank.

25.5.2 Model Analysis

A set of system test data which is referred to as Phase 2 data set [22] is used to illustrate the model performance in this subsection. In this data set, the number of faults detected in each week of testing is found and the cumulative number of faults since the start of testing is recorded for each week. This data set provides the cumulative number of faults by each week up to 21 weeks.

Table 25.8 summarizes the result as well as the ranking of nine selected models from Table 25.1 based on MSE, PRR, PP, and NCD criteria. It is worth to note that one can use the NCD criterion to help in selecting the best model from among model candidates. Table 25.8 shows the NCDs and its corresponding ranking for $w_1 = 2$, $w_2 = 1.5$, and $w_3 = 1$ with respect to MSE, PRR, and PP, respectively.

Table 25.8 [28, page 1488]: Parameter estimation and model comparison when $w_1 = 2$, $w_2 = 1.5$, $w_3 = 1$

Model/criteria	MSE (Rank)	PRR (Rank)	PP (Rank)	NCD value (D_k)	Model rank
1. G-O Model	6.61 (7)	0.69 (1)	1.10 (7)	0.1366837	6
2. Delayed S-shaped	3.27 (5)	44.27 (8)	1.43 (8)	0.1511403	7
3. Inflection S-shaped	1.87 (2)	5.94 (5)	0.90 (4)	0.0801189	2
4. Yamada imperfect debugging model	4.98 (6)	4.30 (4)	0.81 (3)	0.1022629	5
5. PNZ model	1.99 (3)	6.83 (7)	0.96 (6)	0.0855598	4
6. Pham-Zhang model	2.12 (4)	6.79 (6)	0.95 (5)	0.0855490	3
7. Dependent-parameter model (model 1)	43.69 (9)	601.34 (9)	4.53 (9)	1.3395573	9
8. Dependent-parameter model with $m(t_0) \neq 0$, $t_0 \neq 0$ (model 2)	24.79 (8)	1.14 (2)	0.73 (1)	0.3893950	8
9. Vtub-shaped fault-detection rate model	1.80 (1)	2.06 (3)	0.77 (2)	0.0692175	1

Based on the results as shown in Table 25.8, the Vtub-shaped fault-detection rate model seems to provide the best fit based on the normalized criteria distance method.

References

- Pham, H., Zhang, X.: A software cost model with warranty and risk costs. *IEEE Trans. Comput.* **48**, 71–75 (1999)
- Pham, H., Normann, L., Zhang, X.: A general imperfect debugging NHPP model with S-shaped fault detection rate. *IEEE Trans. Reliab.* **48**, 169–175 (1999)
- Goel, A.L., Okumoto, K.: Time-dependent error-detection rate model for software and other performance measures. *IEEE Trans. Reliab.* **28**, 206–211 (1979)
- Ohba, M.: Software reliability analysis models. *IBM J. Res. Dev.* **28**, 428–443 (1984)
- Pham, H.: *Software Reliability*. Springer, London (2000)
- Yamada, S., Ohba, M., Osaki, S.: S-shaped reliability growth modeling for software error detection. *IEEE Trans. Reliab.* **33**, 475–484 (1984)
- Zhang, X., Teng, X., Pham, H.: Considering fault removal efficiency in software reliability assessment. *IEEE Trans. Syst. Man Cybern. A*. **33**, 114–120 (2003)
- Pham, H., Zhang, X.: NHPP software reliability and cost models with testing coverage. *Eur. J. Oper. Res.* **145**, 443–454 (2003)
- Zhang, X., Pham, H.: Predicting operational software availability and its applications to telecommunication systems. *Int. J. Syst. Sci.* **33**(11), 923–930 (2002)
- Pham, H., Wang, H.: A quasi renewal process for software reliability and testing costs. *IEEE Trans. Syst. Man Cybern. A*. **31**, 623–631 (2001)
- Zhang, X., Shin, M.-Y., Pham, H.: Exploratory analysis of environmental factors for enhancing the software reliability assessment. *J. Syst. Softw.* **57**, 73–78 (2001)
- Pham, L., Pham, H.: A Bayesian predictive software reliability model with pseudo-failures. *IEEE Trans. Syst. Man Cybern. A*. **31**(3), 233–238 (2001)
- Zhang, X., Pham, H.: Comparisons of nonhomogeneous Poisson process software reliability models and its applications. *Int. J. Syst. Sci.* **31**(9), 1115–1123 (2000)
- Pham, H.: Software reliability and cost models: perspectives, comparison and practice. *Eur. J. Oper. Res.* **149**, 475–489 (2003)
- Yang, B., Xie, M.: A study of operational, testing reliability in software reliability analysis. *Reliab. Eng. Syst. Safety*. **70**, 323–329 (2000)
- Zhang, X., Jeske, D., Pham, H.: Calibrating software reliability models when the test environment does not match the user environment. *Appl. Stoch. Model. Bus. Ind.* **18**, 87–99 (2002)
- Li, Q., Pham, H.: A generalized software reliability growth model with consideration of the uncertainty of operating environments. *IEEE Access*. **7** (2019)
- Teng, X., Pham, H.: A software cost model for quantifying the gain with considerations of random field environment. *IEEE Trans. Comput.* **53**, 3 (2004)
- Kapur, P.K., Pham, H., Aggarwal, A.G., Kaur, G.: Two dimensional multi-release software reliability modeling and optimal release planning. *IEEE Trans. Reliab.* **61**(3), 758–768 (2012)
- Kapur, P.K., Pham, H., Anand, S., Yadav, K.: A unified approach for developing software reliability growth models in the presence of imperfect debugging and error generation. *IEEE Trans. Reliab.* **60**(1), 331–340 (2011)
- Li, Q., Pham, H.: NHPP software reliability model considering the uncertainty of operating environments with imperfect debugging and testing coverage. *Appl. Math. Model.* **51**(11), 68–85 (2017)
- Pham, H.: *System Software Reliability*. Springer (2006)
- Pham, H.: Software reliability assessment: imperfect debugging and multiple failure types in software development. EG&G-RAAM-10737; Idaho National Engineering Laboratory (1993)
- Pham, H.: A software cost model with imperfect debugging, random life cycle and penalty cost. *Int. J. Syst. Sci.* **27**(5), 455–463 (1996)
- Pham, H., Zhang, X.: An NHPP software reliability model and its comparison. *Int. J. Reliab. Qual. Saf. Eng.* **4**(3), 269–282 (1997)
- Pham, H.: An imperfect-debugging fault-detection dependent-parameter software. *Int. J. Autom. Comput.* **4**(4), 325–328 (2007)
- Pham, H.: A software reliability model with vtub-shaped fault-detection rate subject to operating environments. In: Proc. 19th ISSAT Int'l Conf. on Reliability and Quality in Design, Hawaii (2013)
- Pham, H. (2014), “A new software reliability model with Vtub-shaped fault-detection rate and the uncertainty of operating environments,” *Optimization*, v 63, p. 1481–1490
- Pham, H., Pham, D. H., Pham, H. (2014), “A new mathematical logistic model and its applications,” *Int. J. Inform.Manag. Sci.*, v, 25, no. 2, p. 79–99
- Pham, H.: Loglog fault-detection rate and testing coverage software reliability models subject to random environments. *Vietnam J. Comput. Sci.* **1**(1) (2014)
- Pham, L., Pham, H.: Software reliability models with time-dependent hazard function based on Bayesian approach. *IEEE Trans. Syst. Man Cybern. A*. **30**(1), 25–35 (2000)
- Pham, H.: A generalized fault-detection software reliability model subject to random operating environments. *Vietnam J. Comput. Sci.* **3**(3), 145–150 (2016)
- Sgarbossa, F., Pham, H.: A cost analysis of systems subject to random field environments and reliability. *IEEE Trans. Syst. Man Cybern. C*. **40**(4), 429–437 (2010)
- Zhang, X., Pham, H.: Software field failure rate prediction before software deployment. *J. Syst. Softw.* **79**, 291–300 (2006)

35. Zhu, M., Pham, H.: A two-phase software reliability modeling involving with software fault dependency and imperfect fault removal. *Comput. Lang. Syst. Struct.* **53**(2018), 27–42 (2018)
36. Lee, D.H., Chang, I.-H., Pham, H.: Software reliability model with dependent failures and SPRT. *Mathematics*. **8**, 2020 (2020)
37. Zhu, M., Pham, H.: A generalized multiple environmental factors software reliability model with stochastic fault detection process. *Ann. Oper. Res.* (2020) (in print)
38. Zhu, M., Pham, H.: A novel system reliability modeling of hardware, software, and interactions of hardware and software. *Mathematics*. **7**(11), 2019 (2019)
39. Song, K.Y., Chang, I.-H., Pham, H.: A testing coverage model based on NHPP software reliability considering the software operating environment and the sensitivity analysis. *Mathematics*. **7**(5) (2019). <https://doi.org/10.3390/math7050450>
40. Sharma, M., Pham, H., Singh, V.B.: Modeling and analysis of leftover issues and release time planning in multi-release open source software using entropy based measure. *Int. J. Comput. Syst. Sci. Eng.* **34**(1) (2019)
41. Pham, T., Pham, H.: A generalized software reliability model with stochastic fault-detection rate. *Ann. Oper. Res.* **277**(1), 83–93 (2019)
42. Zhu, M., Pham, H.: A software reliability model incorporating martingale process with gamma-distributed environmental factors. *Ann. Oper. Res.* (2019). <https://doi.org/10.1007/s10479-018-2951-7>
43. Chatterjee, S., Shukla, A., Pham, H.: Modeling and analysis of software fault detectability and removability with time variant fault exposure ratio, fault removal efficiency, and change point. *J. Risk Reliab.* **233**(2), 246–256 (2019)
44. Pham, H.: A logistic fault-dependent detection software reliability model. *J. Univ. Comput. Sci.* **24**(12), 1717–1730 (2018)
45. Song, K.Y., Chang, I.-H., Pham, H.: Optimal release time and sensitivity analysis using a new NHPP software reliability model with probability of fault removal subject to operating environments. *Appl. Sci.* **8**(5), 714–722 (2018)
46. Sharma, M., Singh, V.B., Pham, H.: Entropy based software reliability analysis of multi-version open source software. *IEEE Trans. Softw. Eng.* **44**(12), 1207–1223 (2018)
47. Zhu, M., Pham, H.: A multi-release software reliability modeling for open source software incorporating dependent fault detection process. *Ann. Oper. Res.* **269** (2017). <https://doi.org/10.1007/s10479-017-2556-6>
48. Song, K.Y., Chang, I.-H., Pham, H.: An NHPP software reliability model with S-shaped growth curve subject to random operating environments and optimal release time. *Appl. Sci.* **7**(12), 2017 (2017)
49. Song, K.Y., Chang, I.-H., Pham, H.: A software reliability model with a Weibull fault detection rate function subject to operating environments. *Appl. Sci.* **7**(10), 983 (2017)
50. Li, Q., Pham, H.: A testing-coverage software reliability model considering fault removal efficiency and error generation. *PLoS One.* (2017). <https://doi.org/10.1371/journal.pone.0181524>
51. Zhu, M., Pham, H.: Environmental factors analysis and comparison affecting software reliability in development of multi-release software. *J. Syst. Softw.* **132**, 72–84 (2017)
52. Lee, S.W., Chang, I.-H., Pham, H., Song, K.Y.: A three-parameter fault-detection software reliability model with the uncertainty of operating environment. *J. Syst. Sci. Syst. Eng.* **26**(1), 121–132 (2017)
53. Zhu, M., Pham, H.: A software reliability model with time-dependent fault detection and fault-removal. *Vietnam J. Comput. Sci.* **3**(2), 71–79 (2016)
54. Zhu, M., Zhang, X., Pham, H.: A comparison analysis of environmental factors affecting software reliability. *J. Syst. Softw.* **109**, 150–160 (2015)
55. Chang, I.-H., Pham, H., Lee, S.W., Song, K.Y.: A testing-coverage software reliability model with the uncertainty of operating environments. *Int. J. Syst. Sci.* **1**(4), 220–227 (2014)
56. Cox, D.R.: Regression models and life tables (with discussion). *J. R. Stat. Soc. Ser. B.* **34**, 133–144 (1972)



Hoang Pham is a Distinguished Professor and former Chairman of the Department of Industrial and Systems Engineering at Rutgers University, New Jersey, USA. He is the author or coauthor of 7 books and has published over 200 journal articles, 100 conference papers, and edited 20 books. His numerous awards include the 2009 IEEE Reliability Society *Engineer of the Year Award*. He is a Fellow of the IEEE and IISE.



Xiaolin Teng received his Ph.D. in industrial engineering from Rutgers University in 2001. He also holds master degrees in statistics, computer science, and automation. Currently Dr. Teng works at Meredith Corporation as an associate director. His current research interests include business analytics, machine learning, and data mining.



Contents

26.1	Introduction	496
26.2	Literature Review	496
26.3	General Probabilistic Processes Description	497
26.4	Nonrepairable Degraded Systems Reliability Modeling	497
26.4.1	Degraded Systems Subject to Two Competing Processes.....	497
26.4.2	Systems Subject to Three Competing Processes.....	500
26.4.3	Reliability Evaluation.....	502
26.4.4	Numerical Examples.....	502
26.5	Repairable Degraded Systems Modeling	505
26.5.1	Inspection–Maintenance Model Subject to Two Competing Processes.....	505
26.5.2	Inspection–Maintenance Model for Degraded Systems with Three Competing Processes.....	511
26.6	Complex Systems with Dependent Competing Risks and Random Shocks Using Copulas	516
26.7	Conclusions and Perspectives	517
Appendix A		517
<i>Jacobian Determinant</i>		517
Appendix B		518
References		519

Abstract

The first part of this chapter provides a brief introduction to statistical maintenance modeling subject to multiple failure processes. It includes a description of general probabilistic degradation processes.

The second part discusses detailed reliability modeling for degraded systems subject to competing failure processes without maintenance actions. A generalized multi-state degraded-system reliability model with multiple competing failure processes including degradation processes and random shocks is presented. The operating condition of the multi-state system is characterized by a finite number of states. A methodology to generate the system states when multi-failure processes exist is also discussed. The model can be used not only to determine the reliability of the degraded systems in the context of multi-state functions but also to obtain the probabilities of being in a given state of the system.

The third part describes the inspection–maintenance issues and reliability modeling for degraded repairable systems with competing failure processes. A generalized condition-based maintenance model for inspected degraded systems is discussed. An average long-run maintenance cost rate function is derived based on an expression for degradation paths and cumulative shock damage, which are measurable. An inspection sequence is determined based on the minimal maintenance cost rate. Upon inspection, a decision will be made on whether to perform preventive maintenance or not. The optimum preventive maintenance thresholds for degradation processes and inspection sequences are also determined based on a modified Nelder–Mead downhill simplex method.

The fourth part briefly discusses some dependent competing risk models with various applications subject to multiple degradation processes and random shocks especially using time-varying copulas.

Finally, the last part is given over to the conclusions and a discussion of future perspectives for degraded-system maintenance modeling.

H. Pham (✉)
Department of Industrial and Systems Engineering, Rutgers University, Piscataway, NJ, USA
e-mail: hopham@soe.rutgers.edu

W. Li
Marketing Science, Javelin Direct, Inc., Irving, TX, USA

Keywords

Degradation process · Preventive maintenance · Maintenance policy · Compound Poisson process · Competing risks

26.1 Introduction

Technology advances mean that most new products are, on one hand, more reliable but, on the other hand, very difficult to maintain during the product life cycle. Designers have been challenged to find new, effective approaches to evaluate reliability in a timely fashion and to maintain such systems in an optimum way. This chapter presents reliability and maintenance models for degraded systems subject to competing failure processes. The accuracy of reliability estimation through a degradation model cannot be ensured unless the unit-to-unit initial variation and within-unit degradation-rate variation are considered. This chapter also discusses a generalized random-coefficient degradation process and randomized logistic degradation process to model the degradation.

26.2 Literature Review

As degradation occurs, system performance changes from perfect functioning to complete failure; the binary assumption used to analyze, model, and compute system reliability is relaxed. Using degradation measures to assess reliability has seen some important findings in the literature. *Tomsky* [1] investigated two regression models for detecting degradation reliability. *Nelson* [2] briefly surveyed the application of accelerated degradation. *Lu* [3] introduced a nonlinear mixed-effects model and estimated model parameters in a two-stage way. Recently, multi-state reliability has received considerable attention. *Levitin* [4] extended the reliability importance measures for multi-state systems with different measures of performance. When the multi-state nature of a system is addressed, a better understanding of the system reliability behavior is obtained. Our third new development is to build a methodology based on the formulation of degradation in terms of a finite discrete state.

It is well known that the effectiveness of a system depends on both the quality of its design and manufacturing process as well as the proper inspection–maintenance actions to prevent it from failing. Inspection–maintenance issues are considered in the second part of this chapter.

Maintenance has evolved from a simple model that deals with machinery breakdowns, to time-based preventive maintenance, to today's condition-based maintenance. It is of great importance to avoid the failure of a system during its actual operating; especially, when such failures are dangerous

and costly. Time-based and condition-based maintenance are the two major approaches for maintenance. Condition-based maintenance is often profitable since it can be used to avoid failure occurrence at the lowest cost and to improve the availability and reliability of complex systems. This chapter examines the problem of developing a mathematical maintenance cost model to determine both the optimal inspection interval time and preventive maintenance threshold of degraded systems with competing failure processes subject to a condition-based maintenance policy.

Pham et al. [5] presented a Markov model for predicting the reliability of k -out-of- n systems in which components are subject to multi-stage degradation as well as catastrophic failures. Due to the aging effect, the failure rate of the component will increase. They considered the state-dependent transition rates for the degradation process. *Pham et al.* [6] derived models for predicting the availability and mean lifetime of multistage degraded systems with partial repairs. In some production systems failures are not possible to detect but can only be determined by inspection [7]. Several authors [8–22] have proposed various inspection policies and models for systems with a degradation process. *Grall et al.* [8] studied a system subject to a random deterioration process. They developed a model based on a stationary process to determine both the preventive maintenance threshold and inspection dates that minimized the average long-run cost rate. *Chelbi and Ait-Kadi* [10] addressed optimal inspection strategies for deteriorating equipment subject to preventive and corrective maintenance. *Klutke and Yang* [11] studied the availability of maintained systems subject to both the effects of the degradation and random shocks. They considered the degradation process as a deterministic function of time t and that shocks occurred according to a Poisson process in which the shock magnitudes are independent and identically distributed (iid) random variables. *Pham and Xie* [13] developed a generalized surveillance model consisting of dual, mutually dependent stochastic processes for surveillance systems. Their model can be used to better understand both the inspection process, the repair unit itself, and to provide information that can be used to assist inspectors in scheduling and prioritizing their future inspections.

The choice of the inspection schedule and preventive maintenance threshold(s) obviously has an important influence on the economic performance of the maintenance policy. The inspection dates and the preventive maintenance threshold(s) are two main decision variables. However, in industrial applications of condition-based maintenance, the preventive maintenance threshold is usually decided based upon the recommendation made by the maintenance people and the inspection schedule often appears to be set by little more than a rule of thumb. Because of the lack of appropriate modeling support, the preventive maintenance threshold is likely to be set conservatively and the inspection schedule may be performed more than is perhaps necessary. The need

for a maintenance model with cost consideration is obvious in this case.

The chapter is organized as follows. The basic concepts and a review of maintenance, as well as a brief description of probabilistic processes for the modeling of degradation and random shocks, is discussed in Sect. 26.3. A general reliability model for degraded nonrepairable systems subject to multiple competing processes is discussed in Sect. 26.4. The inspection–maintenance modeling issues and detection policies for degraded repairable systems considering multiple competing processes are described in Sect. 26.5. Several numerical examples are given in Sects. 26.4 and 26.5. Finally in Sect. 26.6, several future research perspectives and conclusions are briefly discussed.

26.3 General Probabilistic Processes Description

We consider three random processes. The first two are used to model degradation, while the third is a compound Poisson process used for modeling random shocks:

1. $Y(t) = A + Bg(t)$ is called a random-coefficient degradation path, where $A > 0$ and $B > 0$ are independent random variables and $g(t)$ is an increasing time-dependent function. The random variable A measures the initial value of degradation due to a different manufacturer, the manufacturing quality control of new items, variable deterioration during storage until the item is put into service, and so forth [14]. Therefore, the initial degradation value A is a random variable. The variable B is the degradation rate ($B > 0$) and represents the variations among the population; $g(t)$ is an increasing function.
2. $Y(t) = \frac{We^{Bt}}{A+e^{Bt}}$ is called a randomized logistic degradation path function where A and B are independent non-negative random variables, and W is a constant. The random variable A represents the initial threshold level of degradation and B describes the rate at which degradation accumulates. It should be noted that $Y(t) = \frac{We^{Bt}}{A+e^{Bt}}$ is an S-shaped curve and describes the degradation process well. It matches the path of the cumulative degradation of many systems in practice. The S-shaped curve reflects an initial run-in period of low usage, followed by a period of steady rate of usage, and finally ending with an increasing rate of use due to the aging of the system. We establish the relationship between the two random variables A and B via some rearrangements as follows:

$$W \frac{e^{Bt}}{A + e^{Bt}} < H \Rightarrow B < \frac{1}{t} \ln \frac{u_1 A}{1 - u_1}, \quad (26.1)$$

where H is a constant and $u_1 = \frac{H}{W}$.

3. Let $D(t) = \sum_{i=0}^{N(t)} X_i$ represent a sequence of random shocks in which each shock causes independent damage X_i to the whole system where the X_i are iid with a probability distribution function (pdf) of $f_X(x)$, and a cumulative distribution function (cdf) of $F_X(x)$; $\{N(t), t \geq 0\}$ is a Poisson process with parameter $\lambda > 0$ that is independent of the sequence $\{X_i\}$; $F_X^{(k)}(x)$ denotes the k -th convolution.

The stochastic process $D(t) = \sum_{i=0}^{N(t)} X_i$ is called a compound Poisson process where $N(t)$ is the number of shocks that have occurred up to time t , X_i is the damage caused by the i -th shock, and $D(t)$ is the cumulative damage up to time t .

26.4 Nonrepairable Degraded Systems Reliability Modeling

This section addresses reliability models for nonrepairable degraded systems. First, we discuss a model for systems subject to two competing processes. Then, we present a generalized situation where systems are subjected to three competing processes.

26.4.1 Degraded Systems Subject to Two Competing Processes

Model Description

The modeling assumptions are as follows:

1. Each system has a state space $\Omega_U = \{M, \dots, 1, 0, F\}$.
2. The system fails either due to degradation ($Y(t) > G$) or catastrophic failure $[D(t) = \sum_{i=1}^{N_2(t)} X_i > S]$. The system may either go from state i to the next degraded state $i - 1$ or directly to the catastrophic failure state F , $i = M, \dots, 1$.
3. No repair or maintenance is performed on the system.
4. Since $Y(t)$ describes the total damage up to time t , it is natural to assume that it is nondecreasing.
5. The two processes $Y(t)$ and $D(t)$ are independent.
6. At time $t = 0$, the system is in state M .

We consider a degradable system suited at a random environment where degradation and random shocks can contribute to an effect of the life of a system. In this section, we discuss the case where systems are subject to two failure processes, called a continuous and increasing degradation process $Y(t)$, and a random shock process $D(t)$. Whichever process occurs first causes the system to failure.

Figure 26.1 illustrates the system flow diagram of the two competing failure processes. In Fig. 26.1, we use either of the random processes described in Sect. 26.3 to represent a degradation process where random shocks are represented by

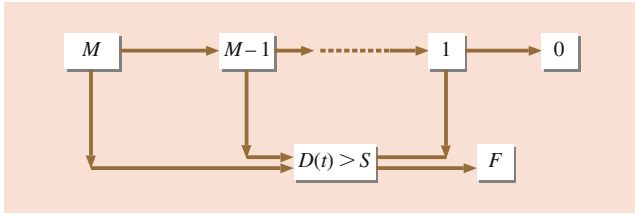


Fig. 26.1 The flow diagram of a system subjected to two competing failure processes

a stationary and independent increment process. Then, we discuss a method to formulate these two processes from a multi-state standing point. That is, suppose that the operating conditions of the system at any point in time could be classified into one of a finite number of the states, say $\Omega_U = \{M, \dots, 1, 0, F\}$. We view the degradation process in terms of a finite number of states. For example, when the value of the degradation process $Y(t)$ falls into a predefined interval then its corresponding state will be determined. Let us define as follows:

$[0, W_M], \dots, (W_2, W_1]$ are the intervals associated with the degradation process where $W_M < W_{M-1} < \dots < W_2 < W_1$. A one-to-one relationship between the element of $\Omega = \{M, \dots, 1, 0\}$ and its corresponding interval is set up as follows:

- when $Y(t) \in [0, W_M] \Rightarrow$ in state M ,
- when $Y(t) \in (W_M, W_{M-1}] \Rightarrow$ in state $M - 1$,
- ⋮
- when $Y(t) \in [W_i, W_{i-1}] \Rightarrow$ in state i ,
- when $Y(t) \in [W_2, W_1] \Rightarrow$ in state 1 ,
- when $Y(t) > W_1 \Rightarrow$ in state 0 .

Reliability Evaluation

The most general situation is to allow each degradation process to be described by a number of different discrete states. We now define the probability in each state. Let $P_i(t)$ be the probability that the value of $Y(t)$ will fall within a predefined interval corresponding to state i with $D(t) \leq S$. From state i , the system will make a direct transition to state $i - 1$ due to gradual degradation, or to catastrophic failure state F due to a random shock.

The reliability function is defined as:

$$\begin{aligned}
 R_M(t) &= P(\text{state} \geq 1) \\
 &= \sum_{i=1}^M P_i(t) \\
 &= P[Y(t) \leq G, D(t) \leq S],
 \end{aligned}
 \tag{26.2}$$

where $P_i(t)$ is the probability of being in state i .

Suppose a system fails if the degradation process crosses some threshold, say G ; or the shock damage process crosses some threshold, say S ; T is defined as:

$$T = \inf [t : Y(t) > G \text{ or } D(t) > S]. \tag{26.3}$$

The mean time to failure is expressed as:

$$\begin{aligned}
 E[T] &= \int_0^\infty P[T > t] dt \\
 &= \int_0^\infty P[Y(t) \leq G, D(t) \leq S] dt \\
 &= \int_0^\infty P[Y(t) \leq G] \sum_{j=0}^\infty \frac{(\lambda_2 t)^j e^{-\lambda_2 t}}{j!} F_X^{(j)}(S) dt
 \end{aligned}$$

or, equivalently, that

$$E[T] = \sum_{j=0}^\infty \frac{F_X^{(j)}(S)}{j!} \int_0^\infty P[Y(t) \leq G] (\lambda_2 t)^j e^{-\lambda_2 t} dt. \tag{26.4}$$

The specific expression for $E[T]$ depends on the probability function $P[Y(t) \leq G]$. Sometimes, it is hard to find a closed-form solution. In this case, one can use a numerical method to solve the problem in (26.4).

The probability density function of the time to failure $f_T(t)$ is as follows:

$$\begin{aligned}
 f_T(t) &= -\frac{d}{dt} R(t) \\
 &= -\frac{d}{dt} \{P[Y(t) \leq G] P[D(t) \leq S]\} \\
 &= -\frac{d}{dt} \left\{ P[Y(t) \leq G] \sum_{j=0}^\infty \frac{(\lambda_2 t)^j e^{-\lambda_2 t}}{j!} F_X^{(j)}(S) \right\} \\
 &= -\sum_{j=0}^\infty \frac{F_X^{(j)}(S)}{j!} \frac{d}{dt} \{P[Y(t) \leq G] (\lambda_2 t)^j e^{-\lambda_2 t}\}
 \end{aligned}$$

Let $F_G(t) = P[Y(t) \leq G]$, then $f_G(t) = \frac{d}{dt} F_G(t)$.

$$\begin{aligned}
 f_T(t) &= -\sum_{j=1}^\infty \frac{F_X^{(j)}(S)}{j!} [f_G(t) (\lambda_2 t)^j e^{-\lambda_2 t} \\
 &\quad + F_G(t) j \lambda_2 (\lambda_2 t)^{j-1} e^{-\lambda_2 t} - \lambda_2 F_G(t) (\lambda_2 t)^j e^{-\lambda_2 t}].
 \end{aligned}
 \tag{26.5}$$

Reliability models

Model 1:

$$\begin{cases} Y(t) = A + Bg(t) \\ D(t) = \sum_{i=0}^{N_2(t)} X_i \end{cases}$$

- case 1 : $A \sim \text{normal}, B \sim \text{normal}$
- case 2 : $A \sim U[0, a], B \sim \text{Exp}(b)$

Model 2:

$$\begin{cases} Y(t) = W \frac{e^{Bt}}{A + e^{Bt}}, \text{ where } A \sim U[0, a], B \sim \text{Exp}(b) \\ D(t) = \sum_{i=0}^{N_2(t)} X_i \end{cases}$$

The two reliability models for the system are depicted in Fig. 26.1. In the following, we will take model 2 as an example to illustrate the results in this section. One can also easily apply it for the model 1.

Assume that the degradation process is described by the function $Y(t) = W \frac{e^{Bt}}{A+e^{Bt}}$, where the two random variables A and B are independent, and A follows a uniform distribution with parameter interval $[0, a]$ and B follows an exponential distribution with parameter $\beta > 0$. In short, $A \sim U[0, a], a > 0$ and $B \sim \text{Exp}(\beta), \beta > 0$.

The probability that the system is in state M is as follows:

$$\begin{aligned}
 P_M(t) &= P \left[Y(t) = W \frac{e^{Bt}}{A+e^{Bt}} \leq W_M, D(t) = \sum_{i=0}^{N_2(t)} X_i \leq S \right] \\
 &= \left[\int_{\forall A} \left(B < \frac{1}{t} \ln \frac{u_1 A}{1-u_1} \mid A = x \right) f_A(x) dx \right] \\
 &\quad \times P \left[D(t) = \sum_{i=0}^{N_2(t)} X_i \leq S \right] \\
 &= \left[1 - \frac{1}{a} \left(\frac{1-u_1}{u_1} \right)^{\frac{\beta}{t}} \left(\frac{t}{t-\beta} \right) \left(a^{1-\frac{\beta}{t}} - 1 \right) \right] \\
 &\quad \times e^{-\lambda_2 t} \sum_{j=0}^{\infty} \frac{(\lambda_2 t)^j}{j!} F_X^{(j)}(S).
 \end{aligned} \tag{26.6}$$

The probability that the system is in state i is calculated as follows:

$$\begin{aligned}
 P_i(t) &= P \left[W_{i+1} < W \frac{e^{Bt}}{A+e^{Bt}} \leq W_i, D(t) = \sum_{i=0}^{N_2(t)} X_i \leq S \right] \\
 &= \left[\int_0^a P \left(\frac{1}{t} \ln \frac{u_i A}{1-u_i} < B \leq \frac{1}{t} \ln \frac{u_{i+1} A}{1-u_{i+1}} \mid A = x \right) f_A(x) dx \right] \\
 &\quad \times e^{-\lambda_2 t} \sum_{j=1}^{\infty} \frac{(\lambda_2 t)^j}{j!} F_X^{(j)}(S) \\
 &= \left\{ \frac{1}{a} \left(\frac{t}{t-\beta} \right) \left(a^{1-\frac{\beta}{t}} \right) \left[\left(\frac{1-u_i}{u_i} \right)^{\frac{\beta}{t}} - \left(\frac{1-u_{i+1}}{u_{i+1}} \right)^{\frac{\beta}{t}} \right] \right\} \\
 &\quad \times e^{-\lambda_2 t} \sum_{j=0}^{\infty} \frac{(\lambda_2 t)^j}{j!} F_X^{(j)}(S),
 \end{aligned} \tag{26.7}$$

where $\mu_i = \frac{W_i}{W}, i = M - 1, \dots, 1$.

Similarly, the probability that the system is in state 0 is as follows:

$$\begin{aligned}
 P_0(t) &= P \left[Y(t) = W \frac{e^{Bt}}{A+e^{Bt}} > G, D(t) = \sum_{i=0}^{N_2(t)} X_i \leq S \right] \\
 &= \left[\frac{1}{a} \left(\frac{1-u_M}{u_M} \right)^{\frac{\beta}{t}} \left(\frac{t}{t-\beta} \right) \left(a^{1-\frac{\beta}{t}} \right) \right] e^{-\lambda_2 t} \\
 &\quad \times \sum_{j=0}^{\infty} \frac{(\lambda_2 t)^j}{j!} F_X^{(j)}(S).
 \end{aligned} \tag{26.8}$$

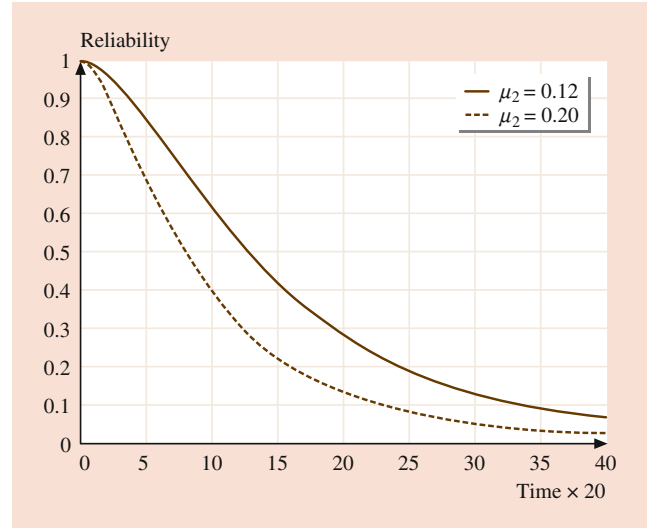


Fig. 26.2 Reliability versus time

The probability for a catastrophic failure state F is given by:

$$\begin{aligned}
 P_F(t) &= P \left[Y(t) = W \frac{e^{Bt}}{A+e^{Bt}} \leq G, D(t) = \sum_{i=0}^{N_2(t)} X_i > S \right] \\
 &= \left[1 - \frac{1}{a} \left(\frac{1-u_1}{u_1} \right)^{\frac{\beta}{t}} \left(\frac{t}{t-\beta} \right) \left(a^{1-\frac{\beta}{t}} \right) \right] \\
 &\quad \times \left[1 - e^{-\lambda_2 t} \sum_{j=0}^{\infty} \frac{(\lambda_2 t)^j}{j!} F_X^{(j)}(S) \right].
 \end{aligned} \tag{26.9}$$

The reliability $R_M(t)$ is expressed as:

$$\begin{aligned}
 R_M(t) &= \sum_{k=1}^M P_k(t) \\
 &= \left[1 - \frac{1}{a} \left(\frac{1-u_M}{u_M a} \right)^{\frac{\beta}{t}} \left(\frac{t}{t-\beta} \right) \left(a^{1-\frac{\beta}{t}} \right) \right] \\
 &\quad \times \left[e^{-\lambda_2 t} \sum_{j=0}^{\infty} \frac{(\lambda_2 t)^j}{j!} F_X^{(j)}(S) \right].
 \end{aligned} \tag{26.10}$$

A Numerical Example

Assume that the degradation is modeled as the function $Y(t) = W \frac{e^{Bt}}{A+e^{Bt}}$ where $A \sim U[0, 5]$ and $B \sim \text{Exp}(10)$. The critical values for the degradation and the shock damage are $G = 500$ and $S = 200$, respectively. The random shocks are measured by the function $D(t) = \sum_{i=1}^{N_2(t)} X_i$, where $X_i \sim \text{Exp}(0.3)$ and X_i s are iid. Figure 26.2 shows the reliability of the system as a function of time, where the solid line represents $N_2(t)$ with $\lambda_2 = 0.12$ and the dotted line represents $N_2(t)$ with $\lambda_2 = 0.20$.

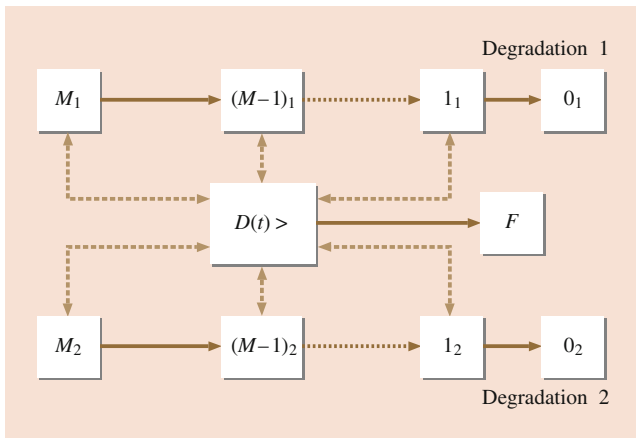


Fig. 26.3 The flow diagram of a system subjected to multiple failure processes [15]

26.4.2 Systems Subject to Three Competing Processes

System Description

In some applications, the systems are subjected to a variety of governing failure processes. In this section, we consider three independent competing failure processes in which two of them are degradation processes (called degradation process 1, which is measured by the function $Y_1(t)$, and degradation process 2, which is measured by $Y_2(t)$), and the third is a random shock process $D(t)$ [15]. Whichever process occurs first causes the system to fail.

Initially, the system is considered to be in its good state (i.e., M_1 and M_2). As time progresses, it can either go to the first degraded state [i.e., $(M-1)_1$ or $(M-1)_2$] upon degradation or can go to a failed state (state F), if subject to random shocks. When a system reaches the first degraded state, it can either stay in that state until the mission time, or it can go to the second degradation state [i.e., $(M-2)_1$ or $(M-2)_2$] upon degradation, or it can go to a failed state (F state) upon random shocks.

The same process will be continued for all stages of degradation except the last degradation, either stage 0_1 or stage 0_2 . If the system reaches the last degradation state, it cannot perform its functions satisfactorily and must be treated as a failure (state 0). Figure 26.3 shows the system flow diagram of the multiple competing transition processes. In Fig. 26.3, the above represents the degradation process 1; the bottom represents the degradation process 2; F represents a catastrophic failure state due to random shocks.

Assumptions

1. The system consists of $(M+2)$ states where state 0 and state F are both complete failure states. State i is a degradation state, $1 \leq i \leq M$.

2. No repair or maintenance is performed on the system.
3. We assume that $Y_i(t)$, $i = 1, 2$ is a nonnegative nondecreasing function at time t , since degradation is an irreversible accumulation of damage.
4. $Y_i(t)$, $i = 1, 2$ and $D(t)$ are statistically independent. The independence assumption implies that the state of one process will have no effect on the state of the others.
5. At time $t = 0$, the system is in state M .
6. The system can fail either due to any of the degradation process when $Y_i(t) > G_i$, $i = 1, 2$ or due to random shocks (in which case it goes to a catastrophic failure state F), i.e., $D(t) = \sum_{i=1}^{N(t)} X_i > S$.
7. The critical threshold value G_i depends upon a function of the states of the degraded systems.

Methodology

In this section, we consider that the degradation paths are modeled by some continuous probabilistic functions. Since the operating condition of the systems is characterized by a finite number of states, let us call the system state space Ω_U . First, we need the discrete continuous processes. In Step 1 below, we discuss a procedure for forcing two degradation processes to become discrete in order to obtain Ω_1 and Ω_2 , which correspond to degradation process 1 and 2, respectively. After we have obtained the degradation process spaces Ω_1 and Ω_2 , we present a methodology for how to establish a relationship between the system state space Ω_U and the degradation and random shock state spaces $\{\Omega_1, \Omega_2, F\}$ in Step 2.

Step 1: Formulate the Degradation Processes in Terms of Discrete State Sets

The two-degradation-process case is considered here. The most general situation is to allow each degradation process to be described by a number of different discrete states. The state space denoted by $\Omega_1 = \{M_1, \dots, 1_1, 0_1\}$ corresponds to degradation process 1 with $M_1 + 1$ states. Similarly, the state space denoted by $\Omega_2 = \{M_2, \dots, 1_2, 0_2\}$ is associated with degradation process 2, having $M_2 + 1$ states. M_1 and M_2 may or may not be the same, and $M_i < \infty$, $i = 1, 2$.

We view the degradation process from the perspective of a finite number of states. For example, when the value of degradation process 1 $Y_1(t)$ falls into a predefined interval, then its corresponding state will be determined. Let us define as follows:

$[0, W_M], \dots, (W_2, W_1]$ are the intervals on the degradation 1 curve (Fig. 26.4a) corresponding to state $M_1, 0_1$, where $W_M < W_{M-1} < \dots < W_1$ and $[0, A_M], \dots, (A_2, A_1]$ are intervals associated with the curve for degradation process 2 (Fig. 26.4b) corresponding to state $M_2, 0_2$, where $A_M < A_{M-1} < \dots < A_1$.

Mathematically, the relationship between the degradation process states $\Omega_1 = \{M_1, \dots, 1_1, 0_1\}$, $\Omega_2 = \{M_2, \dots, 1_2, 0_2\}$

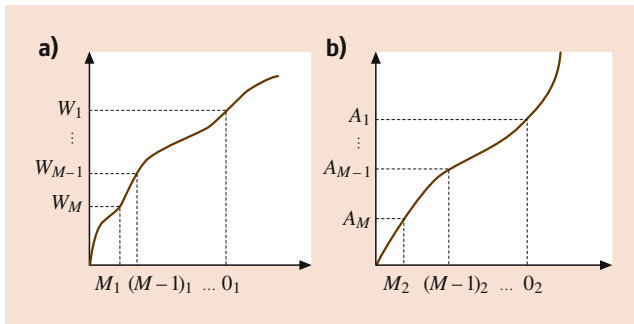


Fig. 26.4 The degradation process functions in multi-state terms for: (a) degradation 1, (b) degradation 2

and their corresponding degradation intervals are given as follows:

$$\begin{aligned}
 &\text{Degradation process 1} \\
 &0 < Y_1(t) \leq W_M : \text{state } M_1 \\
 &W_M < Y_1(t) \leq W_{M-1} : \text{state } (M-1)_1 \\
 &\vdots \\
 &W_2 < Y_1(t) \leq W_1 : \text{state } 1_1 \\
 &G_1 = W_1 < Y_1(t) : \text{state } 0_1
 \end{aligned}$$

$$\begin{aligned}
 &\text{Degradation process 2} \\
 &0 < Y_2(t) \leq A_M : \text{state } M_2 \\
 &A_M < Y_2(t) \leq A_{M-1} : \text{state } (M-1)_2 \\
 &\vdots \\
 &A_2 < Y_2(t) \leq A_1 : \text{state } 1_2 \\
 &G_2 = A_1 < Y_2(t) : \text{state } 0_2
 \end{aligned}$$

Step 2: Generate the System State Space

The system state space is defined as $\Omega_U = \{M, \dots, 1, 0, F\}$, and consists of $M + 2$ states. In this step, we discuss a methodology to develop a function to generate a relationship between the system state space Ω_U and the degradation state spaces $\{\Omega_1, \Omega_2, F\}$. For example, at a given time t , suppose that degradation process 1 is at state $i_1 \in \Omega_1$, and degradation process 2 is at state $j_2 \in \Omega_2$; what is the system state? This question is addressed as follows.

Let us assume that at the current time the system is not in a catastrophic failure state. So state F can be ignored for the time being. Therefore, we can simply look at ways to define a function that has a relationship between Ω and $\{\Omega_1, \Omega_2\}$ instead of Ω_U and $\{\Omega_1, \Omega_2, F\}$.

The operation is described by a mapping function f , which can be written as

$$f : R = \Omega_1 \times \Omega_2 \rightarrow \Omega = \{M, \dots, 1, 0\}$$

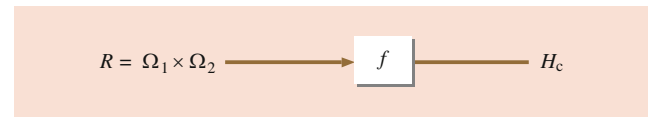


Fig. 26.5 A mapping function

where $R = \Omega_1 \times \Omega_2 = \{(i_1, j_2) \mid i_1 \in \Omega_1, j_2 \in \Omega_2\}$ is a Cartesian product as the input space domain, as shown in Fig. 26.5. The matrix H_c given below is an output space consisting of $M + 1$ elements corresponding to each input-space domain through the function f .

$$H_c = \begin{matrix} & 0_1 & 1_1 & \dots & M_1 \\ \begin{matrix} 0_2 \\ 1_2 \\ \vdots \\ M_2 \end{matrix} & \begin{pmatrix} \times & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \dots & \dots & M \end{pmatrix} \end{matrix}$$

The top row of this matrix H_c represents the state from degradation process 1. The leftmost column represents the state from degradation process 2. The elements of H_c represent $f(i_1, j_2) = k$ where $i_1 \in \Omega_1, j_2 \in \Omega_2$ and $k \in \Omega$. Notice that, in the matrix H_c , all the elements in the first row and first column are zero except that denoted by \times because the system will go to a degraded failure state (state 0) when either of the degradation processes reaches state $0_i, i = 1, 2$. Besides, some elements in the matrix H_c are also zeros since we define that, when degradation 1 is in some low state $l_1 (0_1 < l_1 < M_1)$ and degradation 2 is also in some low state $l_2 (0_2 < l_2 < M_2)$, we consider it a degradation failure. It is also observed that $f(M_1, M_2) = M$, because initially the system is in a brand-new state (perfect state M).

As we mentioned above, the first element in H_c is marked by \times , which means it does not exist. The reason is presented as follows. We define the time to failure as

$$T = \inf [t : Y_1(t) > G_1, Y_2(t) > G_2 \text{ or } D(t) > S]. \quad (26.11)$$

It should be noted that all three processes are competing against each other for the life of a system. However, only one of the three processes (whichever occurs first when its corresponding critical threshold value is exceeded) causing the system to fail. Hence, the following events will not happen:

$$\begin{aligned}
 &P [Y_1(t) > G_1, Y_2(t) > G_2, D(t) \leq S] = 0, \\
 &P [Y_1(t) > G_1, Y_2(t) > G_2, D(t) > S] = 0, \\
 &P [Y_1(t) > G_1, Y_2(t) > G_2, D(t) > S] = 0,
 \end{aligned}$$

and

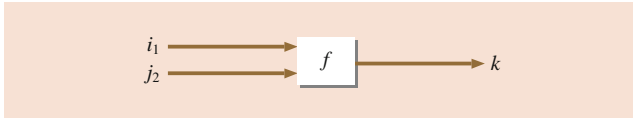


Fig. 26.6 A representation of a system state-generating box

$$P [Y_1(t) < G_1, Y_2(t) > G_2, D(t) > S] = 0,$$

Because $f(0_1, 0_2) = P[Y_1(t) > G_1, Y_2(t) > G_2, D(t) \leq S]$, so the combination of $f(0_1, 0_2)$ does not exist.

The function $f: R = \Omega_1 \times \Omega_2 \rightarrow \Omega = \{M, \dots, 1, 0\}$ is defined with following requirements:

1. $f(0_1, b) = f(a, 0_2) = 0$, where $b \in \Omega_2, a \in \Omega_1, f(M_1, M_2) = M$
2. f is monotonic and nondecreasing in each argument

For instance,

$$\begin{aligned} f(a_1, b_2) &\geq f(l_1, b_2) && \text{if } a_1 \geq l_1, \\ f(a_1, b_2) &\geq f(a_1, l_2) && \text{if } b_2 \geq l_2. \end{aligned}$$

Figure 26.6 demonstrates the system’s state-generating box. There are two inputs i_1 and j_2 and an output k . The inside mapping mechanism is performed by the function f . At time t , suppose that degradation 1 is at state i_1 and degradation 2 is at state j_2 ; i_1 and j_2 are inputs. Via matrix H_c , the system state k is then generated as output.

In the matrix H_c , different state-combination inputs can generate the same results for the system state. To explain this, we need the following definition of the equivalence class.

Definition 26.1

The i -th equivalence class, R_i , is defined as follows:

$$\begin{aligned} R_i &= \{(k_1, j_2) \mid \text{where } k_1 \in \Omega_1, j_2 \in \Omega_2, f(k_1, j_2) = i\}, \\ i &= 0, 1, \dots, M, \end{aligned} \tag{26.12}$$

R_i represents all possible state combinations that generate the system state i ; R_0, \dots, R_M are disjointed sets that partition R into $(M + 1)$ equivalence classes, so that

$$R = \bigcup_{i=0}^M R_i.$$

26.4.3 Reliability Evaluation

In this section, the probability density functions and the system mean time to failure are derived based on the state probabilities given in Sect. 26.4.1. Now, we derive the probability

of being in each state. Initially, the system is in a brand-new state; i.e., in state $M = f_s(R_M)$. The probability for state M is given by

$$P_t(M) = P_t [f_s (R_M)]. \tag{26.13}$$

As defined previously, R_i represents all possible state combinations generating the system state i . The probability of being in state i is the union of all the elements in R_i

$$P_t(i) = P [f_s (R_i)]. \tag{26.14}$$

The probability for a catastrophic failure state F is given by

$$P_t(F) = P [Y_1(t) \leq G_1, Y_2(t) \leq G_2, D(t) > S]. \tag{26.15}$$

The reliability $R(t)$ can be calculated as follows:

$$\begin{aligned} R(t) &= P(\text{system state} \geq 1) \\ &= \bigcup_{i=1}^M P [f_s (R_i)] \\ &= \sum_{i=1}^M P_t(i), \end{aligned} \tag{26.16}$$

where $P_t(i)$ is the probability of being in state i .

The mean time to failure is expressed as [15]:

$$\begin{aligned} E [T] &= \int_0^\infty P (T > t) dt \\ &= \int_0^\infty P [Y_1(t) \leq G_1] P [Y_2(t) \leq G_2] \times \sum_{j=0}^\infty \frac{(\lambda_2 t)^j e^{-\lambda_2 t}}{j!} F_X^{(j)}(S) \end{aligned}$$

or, equivalently, that

$$\begin{aligned} E [T] &= \sum_{j=0}^\infty \frac{F_X^{(j)}(S)}{j!} \int_0^\infty P [Y_1(t) \leq G_1] \times P [Y_2(t) \leq G_2] \\ &\quad (\lambda_2 t)^j e^{-\lambda_2 t} dt. \end{aligned} \tag{26.17}$$

The result in (26.17) obviously would depend on the expression $P[Y_1(t) \leq G_1]P[Y_2(t) \leq G_2]$. The probability density function of time to failure, $f_T(t)$ is therefore as follows:

$$\begin{aligned} f_T(t) &= -\frac{d}{dt} [P (T > t)] \\ &= -\frac{d}{dt} \left\{ P [Y_1 (t \leq G_1)] P [Y_2(t) \leq G_2] \times \sum_{j=0}^\infty \frac{(\lambda_2 t)^j e^{-\lambda_2 t}}{j!} F_X^{(j)}(S) \right\}. \end{aligned} \tag{26.18}$$

26.4.4 Numerical Examples

This example aims to illustrate the results discussed in the previous sections. Consider a system subjected to two degradation processes and random shocks.

Assume that degradation process 1 is described by the function $Y_1(t) = A + Bg(t)$, where the random variables A and B are independent and both follow normal distributions, with mean 90 and variance 2.5, and mean 78 and variance 6, respectively. In short, $A \sim N(90, 2.5)$ and $B \sim N(78, 6)$. The degradation function is assumed to be $g(t) = t^3$. Also $G_1 = 2500$, $W_3 = 1500$, $W_2 = 2000$, and $W_1 = 2500$.

Assume that degradation process 2 is described by $Y_2(t) = W \frac{e^{BBt}}{AA + e^{BBt}}$, where the random variables AA and BB are independent and follow uniform distributions with parameter interval $[0,100]$ and an exponential distribution with parameter 0.1, respectively. In other words, $AA \sim U[0, 100]$ and $BB \sim \text{Exp}(0.01)$. Also $G_2 = 5000$, $A_2 = 2600$, $A_1 = 5000$, and $W = 7000$. Assume that the random shock is represented by $D(t) = \sum_{i=0}^{N(t)} X_i$ with critical value $S = 200$, where $X_i \sim \text{Exp}(0.1)$ and the X_i are iid.

Assume that the states associated with degradation process 1 and degradation process 2 are, respectively, $\Omega_1 = \{3_1, 2_1, 1_1, 0_1\}$ and $\Omega_2 = \{2_2, 1_2, 0_2\}$. We define the system state space as $\Omega_U = \{3, 2, 1, 0, F\}$ and the matrix H_c is given as

$$H_c = \begin{matrix} & \begin{matrix} 0_1 & 1_1 & 2_1 & 3_1 \end{matrix} \\ \begin{matrix} 0_2 \\ 1_2 \\ 2_2 \end{matrix} & \begin{pmatrix} \times & 0 & 0 & 0 \\ 0 & 0 & 2 & 3 \\ 0 & 1 & 2 & 3 \end{pmatrix} \end{matrix}.$$

Then we obtain

$$R = \left\{ \begin{matrix} (0_1, 1_2), (0_1, 2_2), (1_1, 0_2), (2_1, 0_2), (3_1, 0_2), \\ (1_1, 1_2), (2_1, 1_2), (3_1, 1_2), (1_1, 2_2), (2_1, 2_2), \\ (3_1, 2_2) \end{matrix} \right\}$$

The equivalence classes can be listed as follows:

$$\begin{aligned} R_0 &= \{(0_1, 1_2), (0_1, 2_2), (1_1, 0_2), (2_1, 0_2), (3_1, 0_2), (1_1, 1_2)\}, \\ R_1 &= \{(1_1, 2_2)\}, \\ R_2 &= \{(2_1, 1_2), (2_1, 2_2)\}, \\ R_3 &= \{(3_1, 1_2), (3_1, 2_2)\}, \\ R &= \sum_{i=0}^3 R_i. \end{aligned}$$

According to this expression for H_c , the probability of the system being in state 3 is the sum of the probability $f(3_1, 2_2)$ and of the probability $f(3_1, 1_2)$. That sum is calculated as

$$\begin{aligned} P_t(3) &= P_t[f_s(R_3)] = \Phi\left(\frac{1500 - (90 + 78t)}{\sqrt{2.5 + 6t^6}}\right) \\ &\left[1 - \frac{1}{100}(0.4)^{\frac{0.01}{t}} \times \left(\frac{t}{t-0.01}\right) (0.01)^{1 - \frac{0.01}{t}}\right] \\ &\times e^{-\lambda_2 t} \sum_{j=0}^{\infty} \left(\frac{\lambda_2 t}{j!}\right) F_X^{(j)}(200), \end{aligned} \tag{26.19}$$

where Φ is a standard normal distribution.

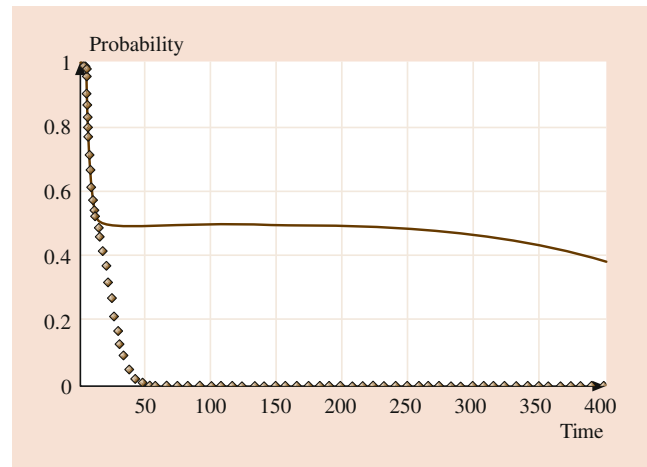


Fig. 26.7 Probability plot for state 3 versus time

Figure 26.7 shows the probability for the system to be in state 3 as a function of time t , where the solid line represents the compound Poisson process $D(t) = \sum_{i=0}^{N(t)} X_i$ with rate $\lambda = 0.04$ and the dotted line represents the compound Poisson process with rate $\lambda = 0.8$. In Fig. 26.7 we observe that, as t progresses to 50, the probability that the system is in state 3 quickly approaches 0 when the rate is given as $\lambda = 0.8$, and is stable with $\lambda = 0.04$.

Because $R_2 = \{(2_1, 1_2), (2_1, 2_2)\}$, the probability of being in state 2 is given by

$$\begin{aligned} P_t(2) &= P_t[f(2_1, 1_2)] + P_t[f(2_1, 2_2)] \\ &= (UV) e^{-\lambda_2 t} \sum_{j=0}^{\infty} \left(\frac{\lambda_2 t}{j!}\right) F_X^{(j)}(200), \end{aligned} \tag{26.20}$$

where

$$U = \Phi\left(\frac{2000 - (90 + 78t)}{\sqrt{2.5 + 6t^6}}\right) - \Phi\left(\frac{1500 - (90 + 78t)}{\sqrt{2.5 + 6t^6}}\right),$$

and

$$V = 1 - \frac{1}{100} \left(\frac{1}{t - 0.01}\right) (0.4)^{\frac{0.01}{t}} \times \left(\frac{t}{t - 0.01}\right) (0.01)^{1 - \frac{0.01}{t}}.$$

Figure 26.8 shows the probability of being in state 2 as a function of time t , where the solid line represents the compound Poisson process $D(t) = \sum_{i=0}^{N(t)} X_i$ with rate $\lambda = 0.04$, and the dotted line represents the compound Poisson process with rate $\lambda = 0.8$. In Fig. 26.8, we observe that, before the time t progresses to 5, the probability of being in state 2 stays close to zero for both rates $\lambda = 0.8$ and $\lambda = 0.04$. It should be noted that the two curves are almost the same for the different values of the rate $\lambda = 0.8$ and $\lambda = 0.04$.

Similarly, the probability of being in state 1 is calculated as:

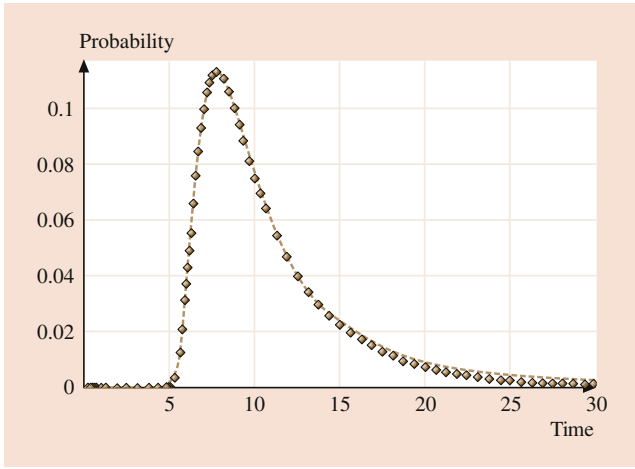


Fig. 26.8 Probability plot for state 2 versus time

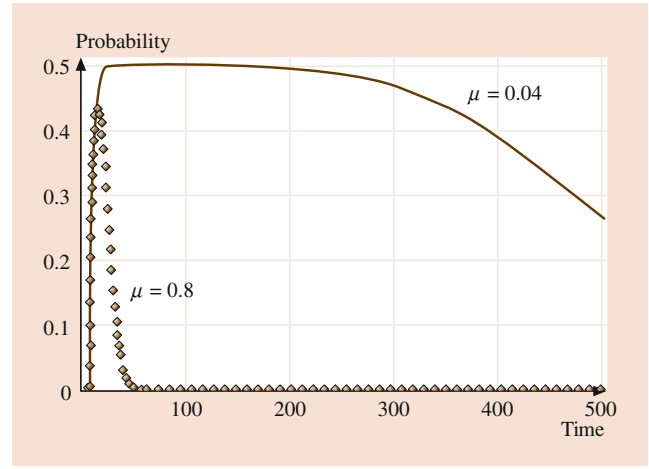


Fig. 26.10 Probability plot for state 0 versus time

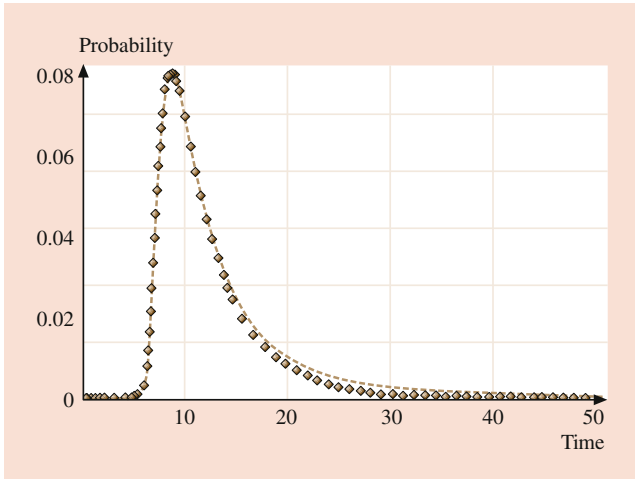


Fig. 26.9 Probability plot for state 1 versus time

$$\begin{aligned}
 P_t(1) &= P_t[f(1_1, 2_2)] \\
 &= E_1 E_2 e^{-\lambda_2 t} \sum_{j=0}^{\infty} \left(\frac{\lambda_2 t}{j!}\right) F_X^{(j)}(200),
 \end{aligned}$$

where $E_1 = \Phi\left(\frac{2500 - (90 + 78t)}{\sqrt{2.5 + 6t^6}}\right) - \Phi\left(\frac{2000 - (90 + 78t)}{\sqrt{2.5 + 6t^6}}\right)$,

$$E_2 = 1 - \frac{1}{100} \left(\frac{t}{t-0.01}\right) \left(\frac{22}{13}\right)^{\frac{0.01}{t}} (0.01)^{1 - \frac{0.01}{t}}. \tag{26.21}$$

Figure 26.9 shows the probability of being in state 1 versus time t , where the solid line represents the compound Poisson process $D(t) = \sum_{i=0}^{N(t)} X_i$ with rate $\lambda = 0.04$, and the dotted line represents the compound Poisson process with rate $\lambda = 0.8$. In Fig. 26.9, we observe that, before the time t progresses to 15, the probability of being in state 1 for both rates $\lambda = 0.8$ and $\lambda = 0.04$ are about the same.

We can also easily obtain the probability of being in state 0 as follows:

$$\begin{aligned}
 P_t(0) &= P[f(0_1, 1_2) + f(0_1, 2_2) + f(1_1, 0_2) \\
 &\quad + f(2_1, 0_2) + f(3_1, 0_2) + f(1_1, 1_2)] \\
 &= (X_1 Y_1 + X_2 Y_2 + X_3 Y_3) e^{-\lambda_2 t} \\
 &\quad \times \sum_{j=0}^{\infty} \left(\frac{\lambda_2 t}{j!}\right) F_X^{(j)}(200),
 \end{aligned}$$

where $X_1 = 1 - \Phi\left(\frac{2500 - (90 + 78t)}{\sqrt{2.5 + 6t^6}}\right)$,

$$X_2 = \Phi\left(\frac{2500 - (90 + 78t)}{\sqrt{2.5 + 6t^2}}\right),$$

$$Y_1 = 1 - \frac{1}{100} (0.4)^{\frac{0.01}{t}} \left(\frac{t}{t-0.01}\right) \times (0.01)^{1 - \frac{0.01}{t}},$$

$$Y_2 = \frac{1}{100} (0.4)^{\frac{0.01}{t}} \left(\frac{t}{t-0.01}\right) \times (0.01)^{1 - \frac{0.01}{t}},$$

$$\begin{aligned}
 X_3 &= \Phi\left(\frac{2500 - (90 + 78t)}{\sqrt{2.5 + 6t^6}}\right) \\
 &\quad - \Phi\left(\frac{2000 - (90 + 78t)}{\sqrt{2.5 + 6t^6}}\right),
 \end{aligned}$$

$$\begin{aligned}
 Y_3 &= 1 - \frac{1}{100} \left[\left(\frac{22}{13}\right)^{\frac{0.01}{t}} + (0.4)^{\frac{0.01}{t}} \right] \\
 &\quad \times \left(\frac{t}{t-0.01}\right) (0.01)^{1 - \frac{0.01}{t}}.
 \end{aligned} \tag{26.22}$$

Figure 26.10 shows the probability that the system is in state 0 versus the time t , where the solid line represents the compound Poisson process $D(t) = \sum_{i=0}^{N(t)} X_i$ with rate $\lambda = 0.04$, and the dotted line represents the compound Poisson process with rate $\lambda = 0.8$. In Fig. 26.10, we observe that the

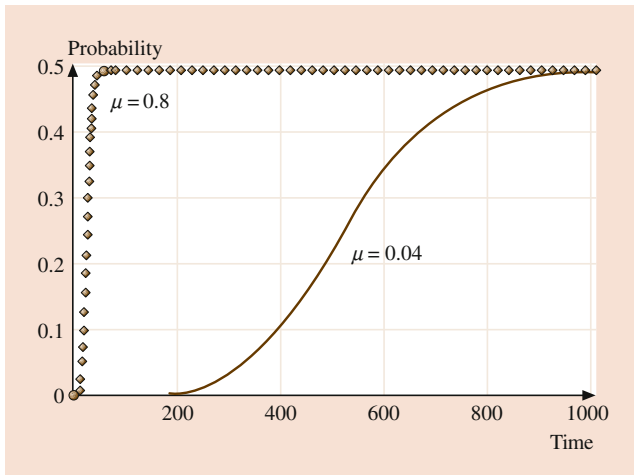


Fig. 26.11 Probability plot for state F versus time

probability of being in state 0 is close to zero when $t > 100$ for the rate $\lambda = 0.8$.

The probability of being in state F is calculated as:

$$P_t(F) = P[Y_1(t) \leq G_1, Y_2(t) \leq G_2, D(t) > S]$$

$$= KL \left[1 - e^{-\lambda_2 t} \sum_{j=0}^{\infty} \left(\frac{\lambda_2 t}{j!} \right) F_X^{(j)}(200) \right],$$

where $K = \Phi \left(\frac{2500 - (90 + 78t)}{\sqrt{2.5 + 6t^6}} \right)$,

$$L = 1 - \frac{1}{100} (0.4)^{\frac{0.01}{t}} \left(\frac{t}{t - 0.01} \right) \left(0.01^{1 - \frac{0.01}{t}} \right). \tag{26.23}$$

Figure 26.11 shows the probability of being in state F as a function of time t , where the solid line represents the compound Poisson process $D(t) = \sum_{i=0}^{N(t)} X_i$ with rate $\lambda = 0.04$, and the dotted line represents the compound Poisson process with rate $\lambda = 0.8$.

Finally, the system reliability $R(t)$ is given by

$$R(t) = P(\text{system state} \geq 1)$$

$$= \sum_{i=1}^3 P_t(i)$$

$$= X_3 Y_3 e^{-\lambda_2 t} \sum_{j=0}^{\infty} \left(\frac{\lambda_2 t}{j!} \right) F_X^{(j)}(200),$$

where $X_3 = \Phi \left(\frac{2000 - (90 + 78t)}{\sqrt{2.5 + 6t^6}} \right)$

$$\times \left\{ 1 - \frac{1}{100} \left[(0.4)^{\frac{0.01}{t}} + \left(\frac{22}{13} \right)^{\frac{0.01}{t}} \right] \right\}$$

$$\times \left(\frac{t}{t - 0.01} \left(0.01^{1 - \frac{0.01}{t}} \right) \right) \Bigg\},$$

$$Y_3 = \left[\Phi \left(\frac{2500 - (90 + 78t)}{\sqrt{2.5 + 6t^6}} \right) - \Phi \left(\frac{2000 - (90 + 78t)}{\sqrt{2.5 + 6t^6}} \right) \right]$$

$$\times \left[1 - \frac{1}{100} \left(\frac{t}{t - 0.01} \right) \right]$$

$$\times \left(\frac{22}{13} \right)^{\frac{0.01}{t}} 0.01^{1 - \frac{0.01}{t}}. \tag{26.24}$$

Figure 26.12 shows the system reliability versus time t , where the solid line represents the compound Poisson process with rate $\lambda = 0.04$, and the dotted line represents the compound Poisson process with rate $\lambda = 0.8$. As for the rate $\lambda = 0.8$ we observe that the system will probably fail after a time t of 50. It seems that the random shock process governs the behavior of the reliability function. Therefore, the dotted line quickly approaches the failure caused by the shock damage.

26.5 Repairable Degraded Systems Modeling

26.5.1 Inspection–Maintenance Model Subject to Two Competing Processes

Model Description

Assumptions

The system starts in a new condition. The assumptions are as follows [22]:

1. The system is not continuously monitored, its state can be detected only by inspection, but system failure is self-announcing without inspection.
2. After a PM or CM action, the system will be restored back to an as-good-as-new state.
3. A CM action is more costly than a PM, and a PM costs much more than an inspection. This implies $C_c > C_p > C_i$.
4. The two processes $Y(t)$ and $D(t)$ are independent.
5. Repair time is not negligible.

Although continuous monitoring processes are feasible for some systems, the cost to monitor the process and the labor required would, however, not make it realistic in practice. Therefore, we need to improve the system performance by determining the periodic inspections with maintenance action that will minimize the average total system maintenance

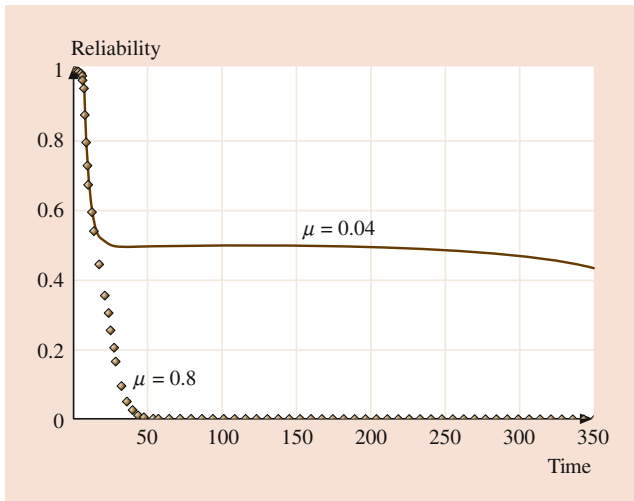


Fig. 26.12 Reliability versus time

cost. Since system deterioration while running leads to system failure, it proves better to assume that the degradation paths are continuous and increasing functions.

Inspection–Maintenance Policy

It is proposed that the system is periodically inspected at times $\{I, 2I, \dots, nI, \dots\}$. We assume that the degradation $(\{Y(t)\}_{t \geq 0})$ and random shock processes $(\{D(t)\}_{t \geq 0})$ are independent. Let T denote the time to failure, defined as

$$T = \inf [t > 0 : Y(t) > G \text{ or } D(t) > S],$$

where G is the critical value for $\{Y(t)\}_{t \geq 0}$ and S is the threshold level for $\{D(t)\}_{t \geq 0}$.

The two threshold values L and G (where G is fixed) effectively divide the system state into three regions, as illustrated in Fig. 26.13. They are: the doing-nothing zone; the PM zone; and the CM zone. The maintenance action will be performed when either of the following situations occurs.

1. The current inspection reveals that the system condition falls into the PM zone, and this state was not found on previous inspection. At inspection time iI , the system falls into the PM zone, which means $\{Y[(i - 1)I] \leq L, D[(i - 1)I] \leq S\} \cap \{L < Y(iI) \leq G, D(iI) \leq S\}$. Then PM action is performed and will take a random time R_1 .
2. When the system fails at T , a CM action is taken immediately and takes time R_2 .

It is assumed that both PM and CM actions are considered to be perfect. Even though both PM and CM actions bring the system back to an as-good-as-new state, they are, physically, not necessarily the same, since a CM has to performed on

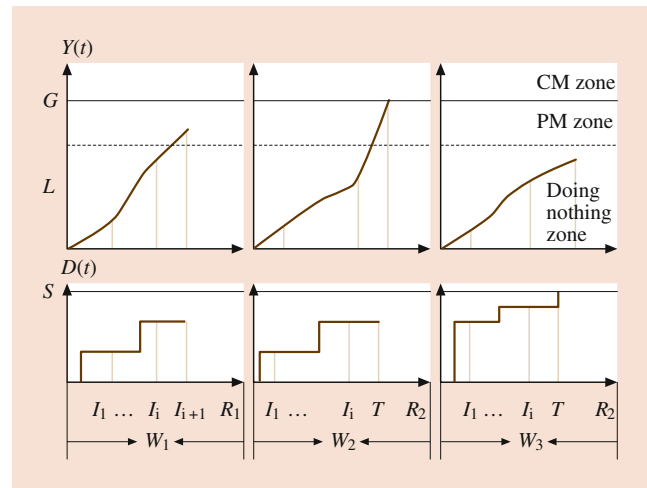


Fig. 26.13 The evolution of the system

a worse system. Hence, CM is likely to be more complex and expensive. Therefore, it is realistic to assume that the repair time is not negligible. This chapter considers that the PM action will take a random amount of time R_1 and that a CM action will take a random amount of time R_2 . After a PM or a CM action is performed, the system is renewed. A new sequence of the inspection would start again, defined in the same way.

Maintenance Cost Modeling

In this section, an explicit expression for the average long-run maintenance cost per unit time is derived. The objectives of the model are to determine the optimal PM threshold L and the optimal inspection time I . From the basics of renewal reward theory, we have

$$\lim_{t \rightarrow \infty} \frac{C(t)}{t} = \frac{E[C_1]}{E[W_1]}.$$

We now model the average total maintenance cost per unit time on a single renewal cycle instead of $\lim_{t \rightarrow \infty} \frac{C(t)}{t}$; then we will analyze $E[C_1]$ and $E[W_1]$.

Expected Maintenance Cost Analysis in a Cycle

The expected total maintenance cost during a cycle $E[C_1]$ is expressed as [22]:

$$E[C_1] = C_i E[N_I] + C_p E[R_1] P_p + C_c E[R_2] P_c. \quad (26.25)$$

During a renewal cycle, activities in terms of costs include: inspection cost, time to repair, and PM or CM actions. The renewal cycle will end by either a PM or a CM action. With a probability of P_p , the cycle will end with a PM action and it will take on average an amount of time $E[R_1]$ to complete a PM action, with a corresponding cost of $C_p E[R_1] P_p$.

Similarly, if a cycle ends with a CM action with probability P_c , it will take on average an amount of time $E[R_2]$ to complete a CM action, with a corresponding cost of $C_c E[R_2] P_c$. In the following, we will perform the analysis of $E[C_1]$.

Calculate $E[N_I]$

Let $E[N_I]$ denote the expected number of inspections during a cycle. $E[N_I]$ can be obtained as:

$$E[N_I] = \sum_{i=1}^{\infty} (i) P(N_I = i). \tag{26.26}$$

Obviously $\sum_{i=1}^{\infty} P(N_I = i) = 1$. There will be a total of i inspections during a cycle if the first PM trigger falls within the time interval $[(i - 1)I, iI]$, or if the system condition is in the doing-nothing zone before time iI and the system fails during the interval $[iI, (i + 1)I]$. In other words, the inspection will stop when the i -th inspection finds that a PM condition is satisfied while this situation was not revealed in the previous inspection, or the system fails during the interval $[iI < T \leq (i + 1)I]$ while the system is in the doing-nothing zone before iI .

Let $P(N_I = i)$ denote the probability that there a total of i inspections occur in a renewal cycle. Then we have

$$\begin{aligned} P(N_I = i) &= P\{Y[(i - 1)I] \leq L, D[(i - 1)I] \leq S\} \\ &\quad \times P[L < Y(iI) \leq G, D(iI) \leq S] \\ &\quad + P[Y(iI) \leq L, D(iI) \leq S] \\ &\quad \times P[iI < T \leq (i + 1)I]. \end{aligned} \tag{26.27}$$

Hence,

$$\begin{aligned} E[N_I] &= \sum_{i=1}^{\infty} i \left\{ P\{Y[(i - 1)I] \leq L, D[(i - 1)I] \leq S\} \right. \\ &\quad \times P[L < Y(iI) \leq G, D(iI) \leq S] \\ &\quad + P[Y(iI) \leq L, D(iI) \leq S] \\ &\quad \left. \times P[iI < T \leq (i + 1)I] \right\}. \end{aligned} \tag{26.28}$$

We now calculate the probabilities $P\{Y[(i - 1)I] \leq L, D[(i - 1)I] \leq S\}$ and $P[L < Y(iI) \leq G, D(iI) \leq S]$ with the following two different expressions for $Y(t)$.

A. Assume $Y(t) = A + Bg(t)$ where $A \sim N(\mu_A, \sigma_A^2)$, $B \sim N(\mu_B, \sigma_B^2)$, and A and B are independent. Given $g(t) = t$. $D(t) = \sum_{i=0}^{N(t)} X_i$ where the X_i are iid and $N(t) \sim$

Poisson(λ). Then

$$\begin{aligned} &P\{Y[(i - 1)I] \leq L, D[(i - 1)I] \leq S\} \\ &= P[A + B(i - 1)I \leq L] \\ &\quad \times P\left\{D[(i - 1)I] = \sum_{i=0}^{N[(i-1)I]} X_i \leq S\right\} \\ &= \Phi\left(\frac{L - (\mu_A + \mu_B(i - 1)I)}{\sqrt{\sigma_A^2 + \sigma_B^2((i - 1)I)^2}}\right) e^{-\lambda(i-1)I} \\ &\quad \times \sum_{j=0}^{\infty} \frac{(\lambda(i - 1)I)^j}{j!} F_X^{(j)}(S) \end{aligned} \tag{26.29}$$

and

$$\begin{aligned} &P[L < Y(iI) \leq G, D(iI) \leq S] \\ &= \left[\Phi\left(\frac{G - (\mu_A + \mu_B iI)}{\sqrt{\sigma_A^2 + \sigma_B^2(iI)^2}}\right) \right. \\ &\quad \left. - \Phi\left(\frac{L - (\mu_A + \mu_B iI)}{\sqrt{\sigma_A^2 + \sigma_B^2(iI)^2}}\right) \right] e^{-\lambda iI} \\ &\quad \times \sum_{j=0}^{\infty} \frac{(\lambda iI)^j}{j!} F_X^{(j)}(S). \end{aligned} \tag{26.30}$$

B. Assume $Y(t) = W \frac{e^{\beta t}}{A + e^{\beta t}}$, where W is a constant, $A \sim U[0, a]$, $a > 0$; $B \sim \text{Exp}(\beta)$, $\beta > 0$, A and B are independent. $D(t) = \sum_{i=0}^{N(t)} X_i$ where the X_i are iid and $N(t) \sim \text{Poisson}(\lambda)$. Then

$$\begin{aligned} &P\{Y[(i - 1)I] \leq L, D[(i - 1)I] \leq S\} \\ &= \left[1 - \frac{1}{a} \left(\frac{1-u_1}{u_1}\right)^{\frac{\beta}{i-1}} \left(\frac{(i-1)I}{(i-1)I-\beta}\right) \times \left(a^{1-\frac{\beta}{(i-1)I}} - 1\right) \right] \\ &\quad \times \sum_{j=0}^{\infty} \frac{[\lambda(i-1)I]^j}{j!} F_X^{(j)}(S), \end{aligned} \tag{26.31}$$

where $u_1 = L/W$. Similarly,

$$\begin{aligned} &P[L < Y(iI) \leq G, D(iI) \leq S] \\ &= \left\{ \frac{1}{a} \left(\frac{iI}{iI-\beta}\right) \times \left(a^{1-\frac{\beta}{iI}}\right) \left[\left(\frac{1-u_3}{u_3}\right)^{\frac{\beta}{iI}} - \left(\frac{1-u_2}{u_2}\right)^{\frac{\beta}{iI}} \right] \right\} e^{-\lambda iI} \\ &\quad \times \sum_{j=0}^{\infty} \frac{(\lambda iI)^j}{j!} F_X^{(j)}(S), \end{aligned} \tag{26.32}$$

where $u_2 = G/W, u_3 = L/W$.

Secondly, we discuss the calculation of $P[iI < T \leq (i + 1)I]$. The definition of T is $T = \inf\{t > 0: Y(t) > G \text{ or } D(t) > S\}$. According to the definition, we derive the expression:

$$\begin{aligned} & P[iI < T \leq (i + 1)I] \\ &= P\{Y(iI) \leq L, Y[(i + 1)I] > G\} \\ &\quad \times P\{D[(i + 1)I] \leq S\} \\ &\quad + P\{Y[(i + 1)I] \leq L\} \\ &\quad \times P\{D(iI) \leq S, D[(i + 1)I] > S\}. \end{aligned} \tag{26.33}$$

In (26.33), since $Y(iI)$ and $Y[(i + 1)I]$ are not independent, we could obtain the joint pdf $f_{Y(iI), Y[(i+1)I]}(y_1, y_2)$ in order to compute $P\{Y(iI) \leq L, Y[(i + 1)I] > G\}$. We consider two different expressions for $Y(t)$. The details are as follows:

A. Assume $Y(t) = A + Bg(t)$ where $A > 0$ and $B > 0$ are two independent random variables, and $g(t)$ is an increasing function of time t . Assume that $A \sim f_A(a), B \sim f_B(b)$. Let

$$\begin{cases} y_1 = a + bg(iI) \\ y_2 = a + bg[(i + 1)I] \end{cases} .$$

After simultaneously solving the above equations in terms of y_1 and y_2 , we obtain:

$$\begin{aligned} a &= \frac{y_1 g[(i + 1)I] - y_2 g(iI)}{g[(i + 1)I] - g(iI)} = h_1(y_1, y_2), \\ b &= \frac{y_2 - y_1}{g[(i + 1)I] - g(iI)} = h_2(y_1, y_2). \end{aligned}$$

The Jacobian J is given by

$$J = \begin{vmatrix} \frac{\partial h_1}{\partial y_1} & \frac{\partial h_1}{\partial y_2} \\ \frac{\partial h_2}{\partial y_1} & \frac{\partial h_2}{\partial y_2} \end{vmatrix} = \left| \frac{1}{g(iI) - g[(i + 1)I]} \right| .$$

Then the random vector $\{Y(iI), Y[(i + 1)I]\}$ has a joint continuous pdf as follows

$$\begin{aligned} & f_{Y(iI), Y[(i+1)I]}(y_1, y_2) \\ &= J | f_A[h_1(y_1, y_2)] f_B[h_2(y_1, y_2)] . \end{aligned} \tag{26.34}$$

B. Assume $Y(t) = \frac{We^{At}}{B + e^{At}}$ where $A > 0$ and $B > 0$ are independent. Assume $A \sim f_A(a), B \sim f_B(b)$. Let

$$\begin{cases} y_1 = \frac{We^{aiI}}{b + e^{aiI}} \\ y_2 = \frac{We^{a(i+1)I}}{b + e^{a(i+1)I}} \end{cases} .$$

The solutions for a and b can be easily found from the above equations in terms of y_1 and y_2 as follows:

$$\begin{cases} a = \frac{\ln\left(\frac{y_2(y_1 - W)}{y_1(y_2 - W)}\right)}{I} = h_1(y_1, y_2) \\ b = -\frac{e^{\frac{\ln\left(\frac{y_2(y_1 - W)}{y_1(y_2 - W)}\right)(i+1)I}}(y_2 - W)}{y_2} = h_2(y_1, y_2) \end{cases} .$$

It can be shown that the random vector $\{Y(iI), Y[(i + 1)I]\}$ has a joint density function given by

$$\begin{aligned} & f_{Y(iI), Y[(i+1)I]}(y_1, y_2) \\ &= |J| f_A[h_1(y_1, y_2)] f_B[h_2(y_1, y_2)], \end{aligned} \tag{26.35}$$

where the Jacobian determinant J is given in Appendix A.

As for the term $P\{D(iI) \leq S, D[(i + 1)I] > S\}$ in (26.30), since $D(t) = \sum_{i=0}^{N(t)} X_i$ is a compound Poisson process, the compound Poisson process has a stationary independent increment property. Therefore, the random variables $D(iI)$ and $D[(i + 1)I] - D(iI)$ are independent. Using the Jacobian transformation, the random vector $\{D(iI), D[(i + 1)I] - D(iI)\}$ is distributed in the same way as vector $\{D(iI), D[(i + 1)I]\}$. Note that $D(iI)$ and $D(I_{i+1})$ are independent, therefore,

$$\begin{aligned} & P\{D(iI) \leq S, D[(i + 1)I] > S\} \\ &= P\{D(iI) \leq S\} P\{D[(i + 1)I] > S\}. \end{aligned} \tag{26.36}$$

Calculate P_p

Note that either a PM or CM action will end a renewal cycle. In other words, these two events are mutually exclusive at the renewal time point. As a consequence, $P_p + P_c = 1$. The probability P_p can be obtained as follows:

$$\begin{aligned} P_p &= P(\text{PM ending a cycle}) \\ &= \sum_{i=1}^{\infty} P\{Y[(i - 1)I] \leq L, L < Y(iI) \leq G\} \\ &\quad \times P\{D(iI) \leq S\}. \end{aligned} \tag{26.37}$$

Analysis of Expected Cycle Length

Since the renewal cycle ends either by a PM action with probability P_p or a CM action with probability P_c , the mean cycle length $E[W_1]$ is calculated as follows:

$$\begin{aligned} E[W_1] &= \sum_{i=1}^{\infty} E[(iI + R_1) I_{\text{PM occurs in } [(i-1)I, iI]}] \\ &\quad + E[(T + R_2) 1_{\text{CM occurs}}] \\ &= \left\{ \sum_{i=1}^{\infty} iP \{Y[(i - 1)I] \leq L, D[(i - 1)I] \leq S\} \right. \\ &\quad \left. P[L < Y(iI) \leq G, D(iI) \leq S]\right\} \\ &\quad + E[R_1] P_p + (E[T] + E[R_2]) P_c, \end{aligned} \tag{26.38}$$

where $I_{PM \text{ occurs in}[(i-1)I, iI]}$ and $I_{CM \text{ occurs}}$ are the indicator functions.

The mean time to failure, $E[T]$ is given by [22]:

$$\begin{aligned} E[T] &= \int_0^\infty P\{T > t\} dt \\ &= \int_0^\infty P[Y(t) \leq G, D(t) \leq S] dt \\ &= \int_0^\infty P[Y(t) \leq G] \sum_{j=0}^\infty \frac{(\lambda_2 t)^j e^{-\lambda_2 t}}{j!} F_X^{(j)}(S) dt \end{aligned}$$

or, equivalently:

$$E[T] = \sum_{j=0}^\infty \frac{F_X^{(j)}(S)}{j!} \int_0^\infty P[Y(t) \leq G] (\lambda_2 t)^j e^{-\lambda_2 t} dt \quad (26.39)$$

The expression $E[T]$ depends on the probability $P[Y(t) \leq G]$ and cannot always easily be obtained in closed form.

Optimization of the Maintenance Cost Rate Policy

We determine the optimal inspection time I and PM threshold L such that the long-run average maintenance cost rate $EC(L, I)$ is minimized. Mathematically, we wish to minimize the following objective function [22]:

$$\begin{aligned} EC(L, I) &= \frac{\sum_{i=1}^\infty iP_1P_2}{\{\sum_{i=1}^\infty I_iP_1P_2\} + E[R_1]P_p + E[R_2]P_c} \\ &+ \frac{\sum_{i=1}^\infty iV_i\{P_3P_4 + P_5P_6\}}{\{\sum_{i=1}^\infty I_iP_1P_2\} + E[R_1]P_p + E[R_2]P_c} \\ &+ \frac{C_pE[R_1]\sum_{i=1}^\infty P_1P_2}{\{\sum_{i=1}^\infty I_iP_1P_2\} + E[R_1]P_p + E[R_2]P_c} \\ &+ \frac{C_pE[R_2]\{1 - \sum_{i=1}^\infty P_1P_2\}}{\{\sum_{i=1}^\infty I_iP_1P_2\} + E[R_1]P_p + E[R_2]P_c}, \end{aligned} \quad (26.40)$$

where $I_{i-1} = (i - 1)I$, $I_i = iI$, $I_{i+1} = (i + 1)I$ and $V_i = P[Y(iI) \leq L, D(iI) \leq S]$, $P_1: P[Y(I_{i-1}) \leq L, D(I_{i-1}) \leq S]$, $P_2: P[L < Y(I_i) \leq G, D(I_i) \leq S]$, $P_3: P[Y(I_i) \leq L, Y(I_{i+1}) > G]$, $P_4: P[D(I_{i+1}) \leq S]$, $P_5: P[Y(I_{i+1}) \leq L]$, $P_6: P[D(I_i) \leq S, D(I_{i+1}) > S]$.

This complex objective function is a nonlinear optimization problem, and it is hard to obtain closed-form optimal solutions for L and I . Nelder and Mead [23] introduced a downhill simplex method that does not require the calculation of derivatives. A simplex is the most elementary geometrical scheme that can be formed in n dimensions and has $(n + 1)$ vertices. A brief summary of the steps of the method is: each iteration generates a new vertex for the simplex. If the new point is better than at least one of the existing vertices, it then

replaces the worst vertex. The search direction is generated through reflection, expansion, and contraction operations.

A step-by-step algorithm proposed by Li and Pham [21] based on the Nelder–Mead downhill simplex method is summarized as follows:

- Step 1: choose $(n + 1)$ distinct vertices as an initial set $\{Z^{(1)}, \dots, Z^{(n+1)}\}$. Then calculate the function value $f(Z)$ for $i = 1, 2, \dots, (n + 1)$, where $f(Z) = EC(I, L)$. Put the values $f(Z)$ in an increasing order where $f(Z^{(1)}) = \min [EC(I, L)]$ and $f(Z^{(n+1)}) = \max [EC(I, L)]$. Set $k = 0$.
- Step 2: compute the best- n centroid $X^{(k)} = \frac{1}{n} \sum_{i=1}^n Z^{(i)}$.
- Step 3: use the centroid $X^{(k)}$ in Step 2 to compute the away-from-worst move direction

$$\Delta X^{(k+1)} = X^{(k)} - Z^{(n+1)}.$$

- Step 4: set $\lambda = 1$ and compute $f(X^{(k)} + \lambda \Delta X^{(k+1)})$. If $f(X^{(k)} + \lambda \Delta X^{(k+1)}) \leq f(Z^{(1)})$ then go to Step 5. Otherwise, if $f(X^{(k)} + \lambda \Delta X^{(k+1)}) \geq f(Z^{(n)})$ then go to Step 6. Otherwise, fix $\lambda = 1$ and go to Step 8.
- Step 5: Set $\lambda = 2$ and compute $f(X^{(k)} + 2\Delta X^{(k+1)})$. If $f(X^{(k)} + 2\Delta X^{(k+1)}) \leq f(X^{(k)} + \Delta X^{(k+1)})$ then set $\lambda = 2$. Otherwise, set $\lambda = 1$. Then go to Step 8.
- Step 6: If $f(X^{(k)} + \lambda \Delta X^{(k+1)}) \leq f(Z^{(n+1)})$ then set $\lambda = 1/2$. Compute $f(X^{(k)} + \frac{1}{2} \Delta X^{(k+1)})$. If $f(X^{(k)} + \frac{1}{2} \Delta X^{(k+1)}) \leq f(Z^{(n+1)})$ then set $\lambda = 1/2$ and go to Step 8. Otherwise, set $\lambda = -1/2$ and, if $f(X^{(k)} + \frac{1}{2} \Delta X^{(k+1)}) \leq f(Z^{(n+1)})$, then set $\lambda = -1/2$ and go to Step 8. Otherwise, go to Step 7.
- Step 7: shrink the current solution set toward the best $Z^{(1)}$ by $Z^{(i)} = \frac{1}{2} (Z^{(1)} + Z^{(i)})$, $i = 2, \dots, n + 1$. Compute the new $f(Z^{(2)}), \dots, f(Z^{(n+1)})$, let $k = k + 1$, and return to Step 2.
- Step 8: Replace the worst $Z^{(n+1)}$ by $X^{(k)} + \lambda \Delta X^{(k+1)}$. If $\sqrt{\frac{1}{n+1} \sum_{i=1}^{n+1} [f(Z^{(i)}) - \bar{f}]^2} < 0.5$, where \bar{f} is an average value, then STOP. Otherwise, let $k = k + 1$ and return to Step 2. (It should be noted that the criterion in Step 8 is not unique but will depend on how soon you would like the algorithm to stop when the function values at the vertices are close. Here we do this when the difference between the maximum and the minimum values of f is less than 0.5.)

A Numerical Example

Here we present an example to illustrate the results and the step-by-step application procedure.

Assume that the degradation process is described by $Y(t) = A + Bg(t)$, where A and B are independent and follow a uniform distribution with parameter interval $[0,4]$ and an exponential distribution with parameter 0.3, i.e., $A \sim U(0, 4)$ and $B \sim \text{Exp}(-0.3t)$, respectively, and $g(t) = \sqrt{t}e^{0.005t}$.

Table 26.1 Optimal values I and L

k	$Z^{(1)}$	$Z^{(2)}$	$Z^{(3)}$	Search result
0	(25,20) EC(I, L) = 564.3	(20,18) EC(I, L) = 631.1	(15,10) EC(I, L) = 773.6	(37.5, 38) EC(I, L) = 440.7
1	(37.5,38) EC(I, L) = 440.7	(25,20) EC(I, L) = 564.3	(20,18) EC(I, L) = 631.1	(42.5,40) EC(I, L) = 481.2
2	(37.5,38) EC(I, L) = 440.7	(42.5,40) EC(I, L) = 481.2	(25,20) EC(I, L) = 564.3	(32.5,29) EC(I, L) = 482.2
3	(37.5,38) EC(I, L) = 440.7	(42.5,40) EC(I, L) = 481.2	(32.5,29) EC(I, L) = 482.2	(32.5,33.5) EC(I, L) = 448.9
4	(37.5,38) EC(I, L) = 440.7	(32.5,33.5) EC(I, L) = 448.9	(42.5,40) EC(I, L) = 481.2	(38.75,37.125) EC(I, L) = 441.0
5	(37.5,38) EC(I, L) = 440.7	(38.75,37.125) EC(I, L) = 441.0	(32.5,33.5) EC(I, L) = 448.9	(35.3125,35.25) EC(I, L) = 441.1
6	(37.5,38) EC(I^*, L^*) = 440.7	(38.75,37.125) EC(I, L) = 441.0	(35.3125,35.25) EC(I, L) = 441.4	Stop

Assume that the random shock damage is described by $D(t) = \sum_{i=1}^{N(t)} X_i$, where X_i follows an exponential distribution, i.e., $X_i \sim \text{Exp}(-0.04t)$ and $N(t) \sim \text{Poisson}(0.1)$. Also $G = 50, S = 100, C_i = 900/\text{inspection}, C_c = 5600/\text{CM}, C_p = 3000/\text{PM}, R_1 \sim \text{Exp}(-0.1t)$, and $R_2 \sim \text{Exp}(-0.04t)$. We now determine both the values of I and L so that the average total cost per unit time $EC(I, L)$ is minimized. Following are step-by-step procedure [22]:

- Step 1: since there are two decision variables I and L , we need $(n + 1) = 3$ initial distinct vertices, which are $Z^{(1)} = (25, 20), Z^{(2)} = (20, 18)$, and $Z^{(3)} = (15, 10)$. Set $k = 0$. We calculate the value of $f(Z^{(i)})$ corresponding to each vertex and sort them in increasing order of $EC(I, L)$.
- Step 2: calculate the centroid: $X^{(0)} = (Z^{(1)} + Z^{(1)})/2 = (22.5, 19)$.
- Step 3: generate the search direction: $\Delta X = X^{(0)} - Z^{(2)} = (7.5, 9)$.
- Step 4: set $\lambda = 1$, which will produce a new minimal $EC(30, 28) = 501.76$ that leads us to try an expansion with $\lambda = 2$, that is $(37.5, 38)$.
- Step 5: set $\lambda = 2$. Similarly, calculate $f(Z)$, which leads to $EC(37.5, 38) = 440.7$. Go to Step 8. This result turns out to be a better solution, hence $(15, 10)$ is replaced by $(37.5, 38)$.

The iteration continues and stops at $k = 6$ (Table 26.1) since $\sqrt{\frac{1}{3} \sum_{i=1}^3 [EC(Z^{(i)}) - \overline{EC(I, L)}]^2} < 0.5$, where $\overline{EC(I, L)}$ is the average value.

Table 26.1 illustrates the process of the Nelder–Mead algorithm. In Table 26.1, $Z^{(i)} = (I, L)$. From Table 26.1, we observe that a set of the optimal values is

$$I^* = 37.5, \quad L^* = 38$$

and the corresponding cost value is $EC^*(I, L) = 440.7$.

Table 26.2 The effect of L on P_c for $I = 37.5$

L	P_c
33	0.465
35	0.505
37	0.654
39	0.759

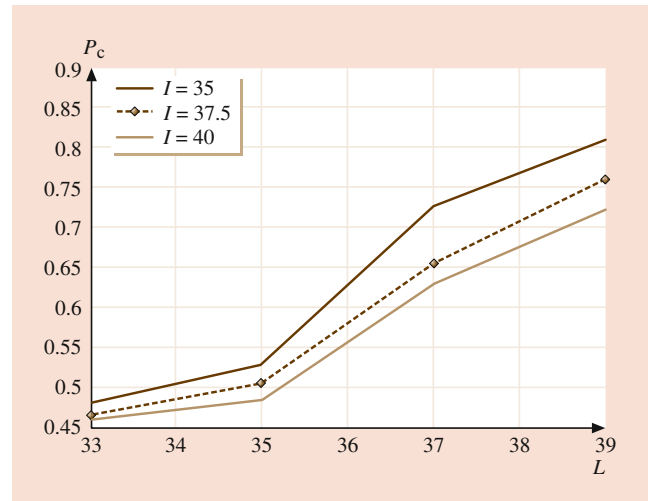


Fig. 26.14 P_c versus L

Table 26.2 illustrates the various values of L on P_c for given $I = 37.5$. From Table 26.2, we observe that the probability P_c increases as L increases. In other words, a larger value for L will put the system at high risk of failure.

Figure 26.14 shows the relationship between L and P_c for different I values, such as $I = 35, I = 37.5$, and $I = 40$. From Fig. 26.14, we observe that P_c is an increasing function of L . This means a higher preventive-maintenance threshold is more likely to result in a failure.

Figure 26.15 depicts the effect of the first inspection time on P_p for various L values such as $L = 33, L = 35, L = 37$, and

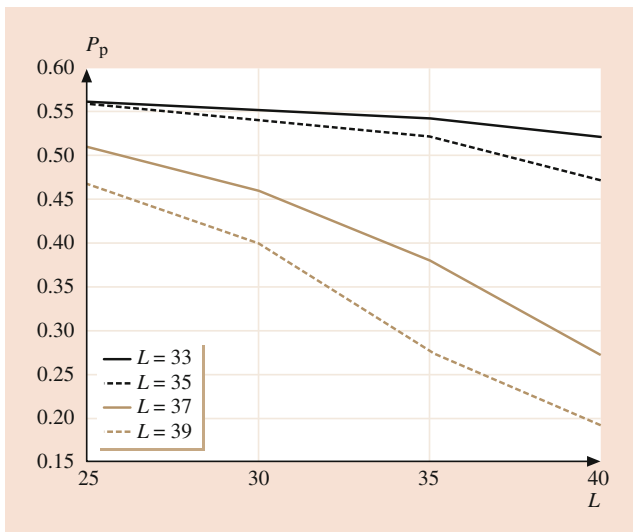


Fig. 26.15 The effect of the inspection sequence on P_p for given L

$L = 39$. Shorter inspection times will cause more-frequent inspection and, as a result, will increase the probability of a PM. From Fig. 26.15, we also observe that, for smaller L values ($L = 33$ and $L = 35$), the curve decreases slightly as I increases; while, for larger values of L such as $L = 37$ and $L = 39$, the curve has a larger decrease as I increases. We also observe that the curve is more sensitive to the value of L , especially when L is large.

In summary, we observe that, on one hand, a lower value of L will result in frequent PM action and prevents full usage of the residual life of the systems. Frequent PM actions might reduce the chance of high deterioration and failures, but will also be costly. On the other hand, a higher L value will keep the system working in a higher-risk condition. Also, frequent inspections will reduce the probability of failure, while incurring additional cost.

26.5.2 Inspection–Maintenance Model for Degraded Systems with Three Competing Processes

General Inspection–Maintenance Description

This section considers systems with inspection-based maintenance subject to three failure processes that are competing for the life of such systems: two of these are degradation processes called degradation process i (measured by $Y_i(t)$ for $i = 1, 2$) and the third is a random shock process measured by the function $D(t)$ [21].

We assume that the three processes are independent and whichever process occurs first will cause the system to fail, where the failure of the system is defined as when $Y_1(t) > G_1$, $Y_2(t) > G_2$ or $D(t) > S$. The state of the system can only be revealed through inspection.

Assumptions

1. System failure is only detected by inspection. Inspections are assumed to be instantaneous, perfect, and nondestructive. Since the system is not continuously monitored, if the system fails it will remain failed until the next inspection, which causes a loss of C_m per unit time. In this case, a maintenance action is begun instantaneously at the inspection time.
2. After a maintenance action, either PM or CM, the system state will start as good as new.
3. A CM action will cost more than a PM action. Similarly, a PM action will cost much more than an inspection itself. This implies that $C_c > C_p > C_i$.
4. The three nondecreasing processes $Y_1(t)$, $Y_2(t)$, and $D(t)$ are independent.
5. No continuous monitoring is performed on the system.
6. The time for a CM or PM action is negligible.

We consider a system subject to three competing processes; two of them are continuous, gradual degradation processes with different characteristics, and the third is a random shock process. Applications of such systems can be found in the Space Shuttle computer complex due to critical mission phases such as boost, reentry, and landing and in electric generator power systems due to the loss of commercial power systems. More related applications can be found in [13].

Although a continuous monitoring process is feasible for some systems, the cost of monitoring the process and the labor required would not make it realistic in practice. Therefore the criteria we consider here is to improve the system performance by performing periodic inspections, with a maintenance action if necessary, to minimize the total system maintenance cost.

Inspection–Maintenance Policy

The length of the inspection will be reduced as the system ages. In other words, the intervals between successive inspections become shorter as the system ages. A geometric sequence is applied in this study to develop the inter-inspection sequence. The inspection time is constructed as $I_n = \sum_{j=1}^n \alpha^{j-1} I_1$, where $0 < \alpha \leq 1$ and I_1 is the first inspection time. We define $U_n = I_n - I_{n-1} = \alpha^{n-1} I_1$ as the inter-inspection interval and $(U_i)_{i \in N}$ as a decreasing geometric sequence. According to the state detected at the inspection I_n , $n = 1, \dots$, one of the following actions will happen [21]:

1. If both degradation values are below their PM thresholds and the shock damage value is less than its threshold, in other words $[Y_1(I_n) \leq L_1, Y_2(I_n) \leq L_2] \cap [D(I_n) \leq S]$, then the system is still in a good condition. In this case, we do

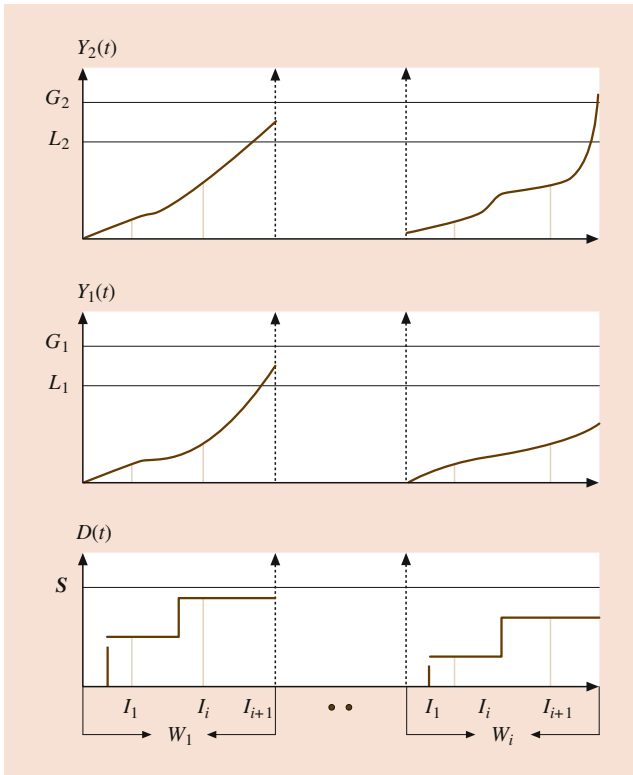


Fig. 26.16 The evolution of the system condition

nothing but determine the next inspection at $I_{n+1} = I_n + U_n$, where U_n is the inter-inspection time between the n -th and the $n + 1$ -th inspection interval.

2. If a degradation process falls into the PM zone [$L_i < Y_i(I_n) \leq G_i, i = 1, 2$] and the other two processes are less than their corresponding critical thresholds, then the system calls for a PM action and it is instantaneously performed accordingly.
3. If any of the process values exceed their corresponding critical thresholds [$Y_i(t) > G_i, i = 1, 2$, or $D(t) > S$], then the system calls for a CM action and it is instantaneously performed. In this case, the system has failed and a CM is performed on the system.

A new sequence of inspection begins, defined in the same way, and the system maintenance follows the same decision rules outlined above. Figure 26.16 shows the evolution of the system, where $Y_1(t)$ and $Y_2(t)$ represent the degradation processes 1 and 2, respectively, and $D(t)$ represents a cumulative shock damage. $(W_i)_{i \in \mathbb{N}}$ is a renewal sequence. Figure 26.17 shows the maintenance zone projected onto the $Y_1(t), Y_2(t)$ planes; G_i and L_i are the CM and PM critical thresholds for $Y_1(t)$, and $Y_2(t)$ respectively.

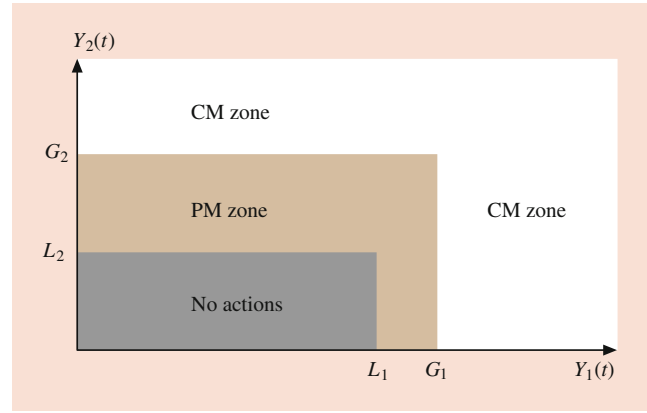


Fig. 26.17 Maintenance zone projected onto $Y_1(t), Y_2(t)$

Maintenance Cost Analysis

The expected total maintenance cost per cycle, $E[C_1]$, is given as:

$$E[C_1] = C_i E[N_I] + C_p P_p + C_c P_c + C_m E[\zeta], \quad (26.41)$$

where C_i is the cost associated with each inspection, C_p is the cost associated with a PM action, and C_c is the CM action cost. Since failure is not self-announcing and it can occur at any given instant time T within the inspection time interval $[I_i, I_{i+1}]$, the system will remain idle during the interval $[T, I_{i+1}]$. The cost coefficient C_m is defined as the penalty cost per unit time associated with such an event.

Calculation

1. Let $P(N_I = i + 1)$ be the probability that there are a total of $(i + 1)$ inspections in the cycle. The expected number of inspections during a cycle, $E[N_I]$, is

$$E[N_I] = \sum_{i=0}^{\infty} (i + 1) P_{i+1}, \quad (26.42)$$

where $P_{i+1} = P(N_I = i + 1)$. Note that

$$P_{i+1} = P(N_I = i + 1) = \bigcup_{j=1}^{17} P(E_j^{(i+1)}),$$

where $E_j^{(i+1)} (j = 1, \dots, 17)$ denotes the renewal cycle that ends at the j -th possible time I_{i+1} . The details of all $E_j^{(i+1)}$, where the $E_j^{(i+1)}$ are mutually disjointed events for $j = 1, \dots, 18$ are listed in Appendix B.

There are a total of 18 system state combinations revealed at any given interval $(I_i, I_{i+1}]$ where there is only one state event, $E_{18}^{(i+1)}$ (Appendix B) representing that the system is in a good condition and that no maintenance

action will be required. Any other remaining state events will trigger either a PM or a CM action at time I_{i+1} .

After some simplifications, we have

$$P_{i+1} = P[Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \\ - P[Y_1(I_{i+1}) \leq L_1, Y_2(I_{i+1}) \leq L_2, D(I_{i+1}) \leq S]. \quad (26.43)$$

Therefore,

$$E[N_1] = \sum_{i=0}^{\infty} (i+1) \{P[Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, \\ D(I_i) \leq S] - P[Y_1(I_{i+1}) \leq L_1, Y_2(I_{i+1}) \leq L_2, \\ D(I_{i+1}) \leq S]\}.$$

2. There will be either a PM or CM action to end a renewal cycle. It is obvious that the two events (PM and CM) are mutually exclusive at the renewal time point: $P_p + P_c = 1$. We now calculate P_p as follows:

$$P_p = P(\text{the cycle ends due to a PM action}) \\ = \sum_{i=0}^{\infty} \sum_{j=1}^3 P[E_j^{(i+1)}].$$

After some simplifications, we obtain

$$P_p = \sum_{i=0}^{\infty} \left\{ P[Y_1(I_i) \leq L_1, L_1 < Y_1(I_{i+1}) \leq G_1] \right. \\ \times P[Y_2(I_i) \leq L_2, Y_2(I_{i+1}) \leq G_2] \\ \times P[D(I_{i+1})] + P[Y_1(I_{i+1}) \leq L_1] \\ \times P[Y_2(I_i) \leq L_2, L_1 < Y_2(I_{i+1}) \leq G_2] \\ \left. \times P[D(I_{i+1})] \right\} \quad (26.44)$$

and $P_c = 1 - P_p$. We can obtain the joint probability density function $f_Y(I_i, Y(I_{i+1}))$ of $Y(I_i)$ and $Y(I_{i+1})$ by computing $P[Y_1(I_i) \leq L_1, Y_1(I_{i+1}) \leq G_1]$ and $P[Y_2(I_i) \leq L_2, Y_2(I_{i+1}) \leq G_2]$.

3. Let T denote the time to failure. That is $T = \inf [t: Y_1(t) > G_1, Y_2(t) > G_2 \text{ or } D(t) > S]$. If $I_i < T \leq I_{i+1}$, the unit will be idle during the interval $[T, I_{i+1}]$. Let $E[\zeta]$ denote the average idle time between the failure occurrence epoch and its inspection during the cycle. Then $E[\xi]$ is calculated as follows:

$$E[\xi] = \sum_{i=0}^{\infty} E[(I_{i+1} - T) 1_{I_i < T \leq I_{i+1}}] \\ = \sum_{j=0}^{\infty} R_j \int_{I_i}^{I_{i+1}} (I_{i+1} - t) dF_T(t), \quad (26.45)$$

where

$$R_j = \left\{ P[Y_1(I_i) \leq L_1, L_1 < Y_1(I_{i+1}) \leq G_1] \right. \\ \times P[Y_2(I_i) \leq L_2, L_1 < Y_1(I_{i+1})] \\ + P[Y_2(I_i) \leq L_2] \\ \times P[Y_1(I_i) \leq L_1, Y_1(I_{i+1}) > G_1] \\ + P[Y_1(I_{i+1}) \leq L_1] \\ \left. \times P[Y_2(I_i) \leq L_2] \right\} P[D(I_i) \leq S]$$

$$F(t) = P[Y_1(t) > G_1, Y_2(t) \leq G_2, D(I_i) \leq S] \\ + P[Y_1(t) \leq G_1, Y_2(t) > G_2, D(I_i) \leq S] \\ + P[Y_1(t) \leq G_1, Y_2(t) \leq G_2, D(I_i) > S]$$

and $1_{I_i < T \leq I_{i+1}}$ is an indicator function.

Expected Cycle Length

The expected cycle length $E[W_1]$ is given as follows:

$$E[W_1] = E[E[W_1|N_1]] \\ = \sum_{i=0}^{\infty} E[W_1|N_1 = i] P(N_1 = i) \\ = \sum_{i=0}^{\infty} I_{i+1} P_{i+1}, \quad (26.46)$$

where P_{i+1} is given in (26.43).

Therefore, the average long-run maintenance cost rate function $EC(L_1, L_2, I_1)$ is a function of the inspection times $\{I_1, \dots, I_i, \dots\}$ and the PM critical threshold values (L_1, L_2) through the functions $P_p, P_c, E[N_1], E[\zeta]$, and $E[W_1]$. The average long-run maintenance cost rate is, in other words, $EC(L_1, L_2, I_1) = \frac{E[C_1]}{E[W_1]}$, and can be obtained by computing the two functions given in (26.41) and (26.46).

Optimization of the Maintenance Cost Rate

The geometric inspection sequence $\{I_1, \dots, I_i, \dots\}$, where $I_n = \sum_{j=1}^n \alpha^{j-1} I_1$, depends on I_1 for given α . In this section, we develop a step-by-step algorithm based on the Nelder–Mead downhill simplex method to obtain the optimum decision variables (I_1, L_1, L_2) such that the long-run average maintenance cost rate $EC(L_1, L_2, I_1)$ is minimized.

Mathematically, the optimization problem of the cost rate function can be formulated as follows [21]:

Optimization Problem

Find I_1, L_1 and $L_2 (0 < L_1 \leq G_1, 0 < L_2 \leq G_2)$ such that

EC (L_1, L_2, I_1)

$$\begin{aligned}
 &= \frac{C_i \sum_{i=0}^{\infty} (i+1) \left\{ P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq S \right] - P \left[Y_1 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq S \right] \right\}}{\sum_{i=0}^{\infty} \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \left\{ P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq S \right] - P \left[Y_1 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq S \right] \right\}} \\
 &+ \frac{C_p \sum_{i=0}^{\infty} \left\{ P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq G_2 \right] P \left[Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2, Y_2 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq G_2 \right] P \left[D \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq S \right] \right\}}{\sum_{i=0}^{\infty} \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \left\{ P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq S \right] - P \left[Y_1 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq S \right] \right\}} \\
 &+ \frac{C_c (1 - \sum_{i=0}^{\infty} \left\{ P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq G_2 \right] P \left[Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2, Y_2 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq G_2 \right] P \left[D \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq S \right] \right\}}{\sum_{i=0}^{\infty} \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \left\{ P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq S \right] - P \left[Y_1 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq S \right] \right\}} \\
 &+ \frac{C_m \sum_{i=0}^{\infty} \left\{ (R_{1i} + R_{2i} + R_{3i}) P \left[D \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq S \right] \int_{\sum_{j=1}^i \alpha^{j-1} I_1}^{\sum_{j=1}^{i+1} \alpha^{j-1} I_1} \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 - t \right) dF_T(t) \right\}}{\sum_{i=0}^{\infty} \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \left\{ P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq S \right] - P \left[Y_1 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_1, Y_2 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq L_2, D \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq S \right] \right\}}
 \end{aligned}$$

is minimum, where

$$\begin{aligned}
 R_{1i} &= P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, L_1 < Y_1 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \leq G_1 \right] \times P \left[Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2, L_1 < Y_1 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \right], \\
 R_{2i} &= P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1, G_1 < Y_1 \left(\sum_{j=1}^{i+1} \alpha^{j-1} I_1 \right) \right] \times P \left[Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2 \right], \\
 R_{3i} &= P \left[Y_1 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_1 \right] \times P \left[Y_2 \left(\sum_{j=1}^i \alpha^{j-1} I_1 \right) \leq L_2 \right].
 \end{aligned}$$

This optimization function is a complex nonlinear function, the optimum solution of which is difficult to find. The Nelder–Mead downhill simplex method (discussed in Sect. 26.5.1) is the most popular direct-search method for obtaining the optimum solution of an unconstrained nonlinear function and does not require the calculation of derivatives.

Numerical Examples

This section illustrates the results in the Sect. 26.5.2. Assume that degradation process 1 is described as the function $Y_1(t) = \frac{We^{B_1 t}}{A_1 + e^{B_1 t}}$, where the random variables A_1 and B_1 are independent and follow a uniform distribution with parameter interval [0,40], and exponential distribution with parameter 1, respectively. In short, $A_1 \sim U[0, 40]$ and $B_1 \sim \text{Exp}(1)$. Similarly, assume that degradation process 2 is modeled as $Y_2(t) = A_2 + B_2 g(t)$ where $A_2 \sim U[0, 2]$, $B_2 \sim \text{Exp}(0.2)$ and $g(t) = \sqrt{t} e^{0.01t}$. Assume that the random shock is represented by the function $D(t) = \sum_{i=0}^{N_2(t)} X_i$, where $X_i \sim \text{Exp}(0.04)$ and $N(t) \sim \text{Poisson}(0.1)$. Also $G_1 = 300$, $G_2 = 70$, and $S = 100$. Assume that the cost parameters are as

follows: $C_c = 560$ units/CM, $C_p = 400$ units/PM, $C_i = 100$ units/inspection, $C_m = 500$ units/unit time and $\alpha = 0.97$. The inspection sequence $\{I_1, \dots, I_n, \dots\}$ is constructed with $I_n = \sum_{j=1}^n \alpha^{j-1} I_1$. We want to determine the values of I_1 and (L_1, L_2) so that the average long-run maintenance cost rate per unit time is minimized. Following are the steps using our proposed algorithm in Sect. 26.5.1:

- Step 1: There are three decision variables, say L_1, L_2 , and I_1 , so we need four distinct vertices as an initial set of values: $Z^{(1)} = (270, 56, 76)$, $Z^{(1)} = (280, 60, 72)$, $Z^{(2)} = (290, 52, 66)$, and $Z^{(3)} = (300, 50, 57)$. Set $k = 0$.

We now calculate the function value $f(Z)$ corresponding to each vertex and put them in increasing order of the objective value $EC(L_1, L_2, I_1)$ from smallest to highest.

- Step 2: Compute the centroid: $X^{(0)} = \frac{1}{3} (Z^{(1)} + Z^{(2)} + Z^{(3)}) = -(-20, 6, 14.3)$.

Table 26.3 Nelder–Mead algorithm results

k	$Z^{(1)} = (L_1, L_2, I_1)$	$Z^{(2)}$	$Z^{(3)}$	$Z^{(4)}$	Search result
0	(270,56,76)	(280,60,72)	(290,52,66)	(300,50,57)	$\lambda = 2$
	$\frac{E[C_1]}{E[W_1]} = 300.7$	$\frac{E[C_1]}{E[W_1]} = 332.2$	$\frac{E[C_1]}{E[W_1]} = 360.4$	$\frac{E[C_1]}{E[W_1]} = 388.2$	$\frac{E[C_1]}{E[W_1]} = 247.9$
1	(240,60,99.9)	(270,56,76)	(280,60,72)	(290,52,66)	$\lambda = 1$
	$\frac{E[C_1]}{E[W_1]} = 247.9$	$\frac{E[C_1]}{E[W_1]} = 300.7$	$\frac{E[C_1]}{E[W_1]} = 332.2$	$\frac{E[C_1]}{E[W_1]} = 360.4$	$\frac{E[C_1]}{E[W_1]} = 248.0$
2	(236,60,99.2)	(240,60,99.9)	(270,56,76)	(280,60,72)	$\lambda = 2$
	$\frac{E[C_1]}{E[W_1]} = 247.9$	$\frac{E[C_1]}{E[W_1]} = 248.0$	$\frac{E[C_1]}{E[W_1]} = 300.7$	$\frac{E[C_1]}{E[W_1]} = 332.2$	$\frac{E[C_1]}{E[W_1]} = 246.7$
3	(187,56,131)	(236,60,99.2)	(240,60,99.9)	(270,56,76)	$\lambda = 1$
	$\frac{E[C_1]}{E[W_1]} = 246.7$	$\frac{E[C_1]}{E[W_1]} = 247.9$	$\frac{E[C_1]}{E[W_1]} = 248.0$	$\frac{E[C_1]}{E[W_1]} = 300.7$	$\frac{E[C_1]}{E[W_1]} = 245.9$
4	(172,60,144)	(187,56,131)	(236,60,99.2)	(240,60,99.9)	Stop
	$\frac{E[C_1]}{E[W_1]} = 245.9$	$\frac{E[C_1]}{E[W_1]} = 246.7$	$\frac{E[C_1]}{E[W_1]} = 247.9$	$\frac{E[C_1]}{E[W_1]} = 248.0$	

- Step 3: Search for the away-from-worst direction: $\Delta X = X^{(0)} - Z^{(4)} = (-20, 6, 14.3)$.
- Step 4: Set $\lambda = 1$, which will generate a new minimal $EC(260, 60, 85.6) = 291.9$ that leads to an expansion with $\lambda = 2$ that is $(240, 60, 99.9)$.
- Step 5: Set $\lambda = 2$. Similarly, compute $f(Z)$, which leads to 247.9. Go to Step 8.

This result turned out to be a better solution, hence $(300, 50, 57)$ is replaced by $(240, 60, 99.9)$.

The iteration continues and stops at $k = 4$ (see Table 26.3) since

$$\sqrt{\frac{1}{4} \sum_{i=1}^4 [EC(Z^{(i)}) - \overline{EC(L_1, L_2, I_1)}]^2} = 0.449 < 0.5,$$

where $\overline{EC(L_1, L_2, I_1)}$ is the average value.

From Table 26.3, we obtain the optimal solution for (L_1, L_2, I_1) as: $(L_1^* = 172, L_2^* = 60, I_1^* = 144)$ and the corresponding average long-run maintenance cost rate is $EC(L_1^*, L_2^*, I_1^*) = 245.9$. Figure 26.18 depicts the average long-run maintenance cost-rate curve $EC(L_1, L_2, I_1)$ as a function of the inspection time interval I_1 for $L_1 = 172$ and $L_2 = 60$.

Table 26.4 presents a sensitivity analysis in terms of the probability that the cycle will end due to a PM action, P_p , for various values of (L_1, L_2) for $\alpha = 0.97$ and $I_1 = 144$. From Table 26.4, we observe that the probability P_p slightly increases as both L_1 and L_2 decrease. This in fact shows that one would perform more PMs than CMs when L_1 and L_2 both become smaller.

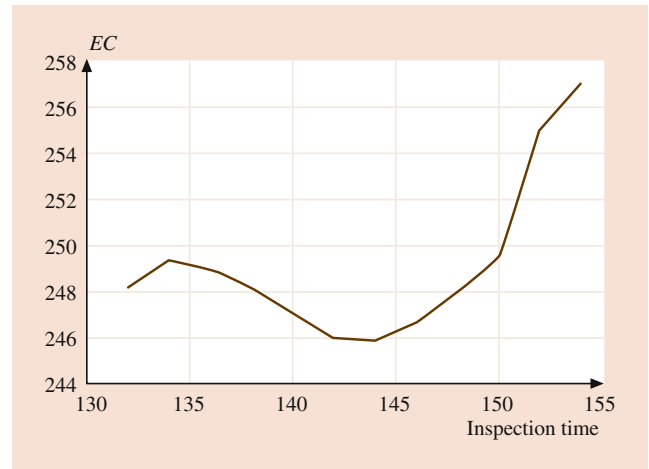


Fig. 26.18 The average maintenance cost $EC(L_1, L_2, I_1)$ versus I_1

Table 26.4 The effect of (L_1, L_2) on P_p for a given inspection sequence

L_1	L_2	P_p
200	60	0.5910
190	58	0.5928
180	56	0.5936
170	54	0.5948
160	52	0.5950
150	50	0.5968

Similarly, Table 26.5 presents the probability that the cycle will end due to a PM action, P_p , for various values of I_1 given $L_1 = 172, L_2 = 60$, and $\alpha = 0.97$. From Table 26.5, we observe that the probability P_p decreases as I_1 increases.

Table 26.5 The effect of the inspection sequence on P_p for fixed PM values

I_1	P_p
110	0.642
120	0.610
130	0.578
140	0.510
150	0.480
160	0.430

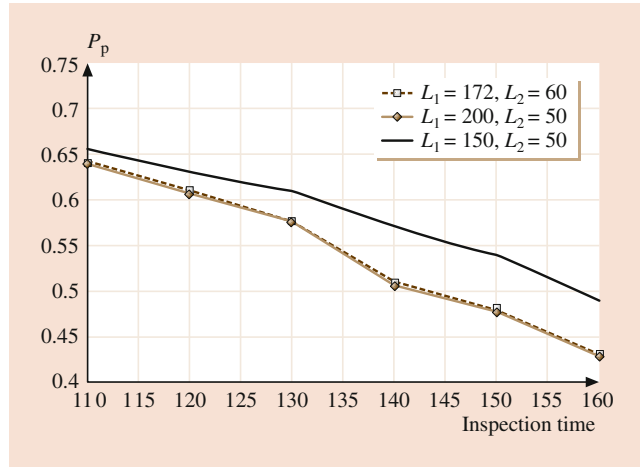


Fig. 26.19 The probability P_p versus I_1 for various pairs (L_1, L_2)

In other words, the maintenance cycle will be more likely to end due to corrective rather than preventive maintenance if one delays inspection. This result can help maintenance managers or inspectors to allocate resources and time.

Figure 26.19 shows the results for the probability that the cycle ends due to PM versus the inspection interval time I_1 for given values of the threshold PM levels (L_1, L_2) . It is interesting to observe from Fig. 26.19 that the results are about the same for the two combinations of $(L_1 = 172, L_2 = 60)$ and $(L_1 = 200, L_2 = 50)$.

26.6 Complex Systems with Dependent Competing Risks and Random Shocks Using Copulas

In this subsection, we discuss briefly some dependent competing risk models for systems subject to multiple degradation processes and random shocks especially using time-varying copulas. Assuming independence between degradation processes may underestimate the entire system reliability [24]. There exist two types of the dependent

configurations that should be taken into consideration: (i) the dependent relationship between degradation processes and random shocks, and (ii) the dependent relationship among various degradation processes.

Taking a human heart as an example, at the age of early 40s, the efficiency of the heart delivering blood to the body will begin to be greatly reduced because of the gradual loss of elasticity of blood vessels. As a result, the arteries may harden or become blocked. Also, many factors can contribute as a shock to the human body, such as non-normal living environment and illness. Diabetes can damage many parts of the human body, such as the heart, kidneys, and blood vessels. Therefore, the human body is a complex system with correlated multiple organs and subsystems that contribute to the proper functioning of the physical mechanism [24].

Finkelstein [25] introduced a generalized Strehler-Mildvan model to estimate the first passage time of the survival function for the system subject to cumulative damage due to biological aging, and sudden killing events. Van Noortwijk et al. [26] put forward a novel approach to combine two stochastic processes of deteriorating resistance and fluctuating load for the reliability analysis of a structural component. Ye et al. [27] proposed a degradation model to capture the two failure mechanisms of degradation and shocks using the Brown-Proshan model under the condition that only failure times and failure modes are recorded without the observable information of shock magnitude and degradation amount. Wang and Pham [28] studied a multi-objective imperfect preventive maintenance policy for complex systems subjected to the dependent competing risks of degradation wear and random shocks, by simultaneously maximizing the system asymptotic availability and minimizing the system cost rate. Sari et al. [29] studied a bivariate degradation model with constant stress to accommodate the dependency between more degradation measures distributed with different marginal functions.

Considerable attention has been paid in the early 2000s to the dependence behavior between random variables modeled by copulas which allows to link the univariate marginal distributions to obtain a joint probability of the events, Cossette et al. [30] studied the discounted penalty function for a generalized Farlie-Bumbel-Morgenstern copula model in the presence of the associations between the claim sizes and inter-claim time in a compound Poisson risk model. Lo and Wilke [31] develop a new copula graphic estimator applied to a model with multiple dependent competing risks, and apply the model to the data set of unemployment duration from Germany. In the work by Miladinovic and Tsokos [32], a modified Gumbel failure model is used to study the system failure time and Bayesian reliability estimates.

The copula method is a flexible, powerful technique to build multivariate distributions which is widely used in various applications including economics, finance, and actuarial science. Kaishev et al. [33] developed a dependent multiple-degradation model to examine the dependencies among causes of death to analyze the impact of complete or partial elimination of causes of death on the survival function from competing risks using copulas. Wang and Pham [24] studied an important classical dependent competing risk model by combining time-scaled covariate factors governed by random shocks into the degradation paths using a copula method. They employed the copula method to build a more flexible dependent competing model for continuously degraded complex systems in which the dependent configuration between random shocks and various degradation processes are linked by time-scaled covariate copula methods. A detail of modeling the dependent competing risks with multiple degradation processes and random shocks of complex systems can be found in [24].

Bedford and Alkali [34] studied a competing risk and opportunistic informative maintenance modeling involving censoring through opportunistic maintenance. Recent studies in the areas of competing risk and dependent failures of systems with various applications can be founded in Liu [35] on the system maintenance modeling for competing risk processes with inter-arrival shocks; in Fang et al. [36] using copula; Palayangoda and Ng [37] considering semiparametric and nonparametric; Donev and Hoffmann [38] on the road transportation research in the presence of data censoring and dependent competing risks; and Austin and Fine [39] and Noordzij et al. [40] about the survival competing risks in medical applications .

26.7 Conclusions and Perspectives

In this chapter, we present reliability and maintenance models for degraded systems with multiple competing failure processes. For mathematical modeling, it is always necessary to make some assumptions in order to make the model applicable in practice. For reliability and maintenance

modeling, assumptions have often played an important role in determining the structure and complexity of the models.

The results of the maintenance models in this chapter can be used to help practitioners and inspectors as well as marketing managers to allocate resources and for the promotion strategies for new products. It would be of interest for future research to implement these results by collecting data and observing product system degradations in practice. Other research problems worth exploring in the future are as follows [22].

1. The objective function discussed in this chapter is to minimize the expected long-run maintenance cost. In practice, costs associated with inspections, preventive maintenance, corrective maintenance, and downtime are sometimes difficult to obtain, even when used in practice. For some critical systems, the overriding goal is to ensure that the system should be available when needed; availability is, therefore, of primary interest, and cost is secondary.

To achieve a high level of availability for a specified inspection rate, it is worth to determine the optimum number of inspections with respect to imperfect repairs, such as minimal and opportunistic schemes, that maximizes the system availability. The time required for imperfect repairs and for replacement policies are random.

2. This chapter assumed that at any time there is unlimited supply of systems available for replacement. In reality, this assumption might not be true due to budget allocation and other constraints. In this case, a random lead time for delivering the new system when needed should be considered. It is essential and practical to analyze the effect of this random lead time on availability. When incorporating random lead time, the expected downtime will increase; therefore, system availability will decrease.

Appendix A

Jacobian Determinant

Below is a 2×2 *Jacobian* determinant

$$J = \frac{y_1 (y_2 - W) \left(\frac{y_2}{y_1(y_2 - W)} - \frac{y_2(y_1 - W)}{y_1(y_2 - W)} \right) [-d(y_1, y_2) - d_1(y_1, y_2) + d_2(y_1, y_2)]}{y_2 (y_1 - W) (I_{i+1} - I_i)} + d_3(y_1, y_2),$$

where

$$d(y_1, y_2) = \frac{\left[\left(\frac{y_1 - W}{y_1(y_2 - W)} - \frac{y_2(y_1 - W)}{y_1(y_2 - W)^2} \right) y_1(y_2 - W)^2 I_{i+1} e^{\left(\frac{\ln\left(\frac{y_2(y_1 - W)}{y_1(y_2 - W)}\right)_{i+1}}{i+1-i} \right)} \right]}{y_2^2 (y_1 - W) (I_{i+1} - I_i)}, y_1 \neq W, y_2 \neq W,$$

$$d_1(y_1, y_2) = \frac{e^{\left(\frac{\ln\left(\frac{y_2(y_1 - W)}{y_1(y_2 - W)}\right)_{i+1}}{i+1-i} \right)}}{y_2}, y_2 \neq W,$$

$$d_2(y_1, y_2) = \frac{e^{\left(\frac{\ln\left(\frac{y_2(y_1 - W)}{y_1(y_2 - W)}\right)_{i+1}}{i+1-i} \right)} (y_2 - W)}{y_2^2}, y_2 \neq W,$$

$$d_2(y_1, y_2) = \frac{d_{31}(y_1, y_2) d_{32}(y_1, y_2)}{y_2^3 (y_1 - W)^2 (I_{i+1} - I_i)^2}, y_1 \neq W,$$

$$d_{31}(y_1, y_2) = \left(\frac{y_1 - W}{y_1(y_2 - W)} - \frac{y_2(y_1 - W)}{y_1(y_2 - W)^2} \right) y_1^2 (y_2 - W)^3, y_2 \neq W,$$

$$d_{32}(y_1, y_2) = \left(\frac{y_2}{y_1(y_2 - W)} - \frac{y_2(y_1 - W)}{y_1^2(y_2 - W)} \right) I_{i+1} e^{\left(\frac{\ln\left(\frac{y_2(y_1 - W)}{y_1(y_2 - W)}\right)_{i+1}}{i+1-i} \right)}, y_2 \neq W.$$

Appendix B

A list of all 18 events:

$$E_1^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 < Y_1(I_{i+1}) \leq G_1, L_2 < Y_2(I_{i+1}) \leq G_2, D(I_{i+1}) \leq S]$$

$$E_2^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 < Y_1(I_{i+1}) \leq G_1, Y_2(I_{i+1}) \leq L_2, D(I_{i+1}) \leq S]$$

$$E_3^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [Y_1(I_{i+1}) \leq L_1, L_2 < Y_2(I_{i+1}) \leq G_2, D(I_{i+1}) \leq S]$$

$$E_4^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 < Y_1(I_{i+1}) \leq G_1, Y_2(I_{i+1}) > G_2, D(I_{i+1}) \leq S]$$

$$E_5^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 < Y_1(I_{i+1}) \leq G_1, L_2 < Y_2(I_{i+1}) \leq G_2, D(I_{i+1}) > S]$$

$$E_6^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 < Y_1(I_{i+1}) \leq G_1, Y_2(I_{i+1}) > G_2, D(I_{i+1}) > S]$$

$$E_7^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [Y_1(I_{i+1}) > G_1, L_2 > Y_2(I_{i+1}), D(I_{i+1}) \leq S]$$

$$E_8^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [Y_1(I_{i+1}) > G_1, L_2 < Y_2(I_{i+1}) \leq G_2, D(I_{i+1}) \leq S]$$

$$E_9^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [Y_1(I_{i+1}) > G_1, L_2 > Y_2(I_{i+1}), D(I_{i+1}) > S]$$

$$E_{10}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [Y_1(I_{i+1}) > G_1, L_2 < Y_2(I_{i+1}) \leq G_2, D(I_{i+1}) > S]$$

$$E_{11}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [Y_1(I_{i+1}) > G_1, Y_2(I_{i+1}) > G_2, D(I_{i+1}) \leq S]$$

$$E_{12}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [Y_1(I_{i+1}) > G_1, Y_2(I_{i+1}) > G_2, D(I_{i+1}) > S]$$

$$E_{13}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 < Y_1(I_{i+1}) \leq G_1, L_2 > Y_2(I_{i+1}), D(I_{i+1}) > S]$$

$$E_{14}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 > Y_1(I_{i+1}), L_2 > Y_2(I_{i+1}), D(I_{i+1}) \leq S]$$

$$E_{15}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 > Y_1(I_{i+1}), L_2 > Y_2(I_{i+1}), D(I_{i+1}) > S]$$

$$E_{16}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 > Y_1(I_{i+1}), L_2 < Y_2(I_{i+1}) \leq G_2, D(I_{i+1}) > S]$$

$$E_{17}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [L_1 > Y_1(I_{i+1}), Y_2(I_{i+1}) > G_2, D(I_{i+1}) \leq S]$$

$$E_{18}^{(i+1)} = [Y_1(I_i) \leq L_1, Y_2(I_i) \leq L_2, D(I_i) \leq S] \cap [Y_1(I_{i+1}) \leq L_1, Y_2(I_{i+1}) \leq L_2, D(I_{i+1}) \leq S]$$

References

1. Tomsy J.: Regression models for detecting reliability degradation. In: Proceedings of Annual Reliability Maintainability Conference, 238–244 (1982)
2. Nelson, W.: Accelerated Testing: Statistical Methods, Test Plans, and Data Analysis. Wiley, New York (1990)
3. Lu H.J.: The use of degradation measures in assessing reliability. Ph.D. thesis, Iowa State University, Ames (1992)
4. Levitin, G.: Reliability of multi-state systems with two failure-modes. *IEEE Trans. Reliab.* **52**, 340–348 (2003)
5. Pham, H., Suprasad, A., Misra, R.B.: Reliability and MTTF prediction of k -out-of- n complex systems with components subjected to multiple stages of degradation. *Int. J. Syst. Sci.* **27**(10), 995–1000 (1996)
6. Pham, H., Suprasad, A., Misra, R.B.: Availability and mean life time prediction of multi-stage degraded system with partial repairs. *Reliab. Eng. Syst. Saf.* **56**, 169–173 (1997)
7. Bris, R., Chatelet, E., Yalaoui, F.: New method to minimize the preventive maintenance cost of series–parallel systems. *Reliab. Eng. Syst. Saf.* **82**, 247–255 (2003)
8. Grall, A., Berenguer, C., Dieulle, L.: A condition-based maintenance policy for stochastically deteriorating systems. *Reliab. Eng. Syst. Saf.* **76**, 167–180 (2002)
9. Grall, A., Dieulle, L., Berenguer, C., Roussignol, M.: Continuous-time predictive-maintenance scheduling for a deteriorating system. *IEEE Trans. Reliab.* **51**(2), 141–150 (2002)
10. Chelbi, A., Ait-Kadi, D.: An optimal inspection strategy for randomly failing equipment. *Reliab. Eng. Syst. Saf.* **63**, 127–131 (1999)
11. Klutke, G.A., Yang, Y.J.: The availability of inspected systems subjected to shocks and graceful degradation. *IEEE Trans. Reliab.* **44**, 371–374 (2002)
12. Zuo, M.J., Liu, B., Murthy, D.N.P.: Replacement–repair policy for multi-state deteriorating products under warranty. *Eur. J. Oper. Res.* **123**, 519–530 (2000)
13. Pham, H., Xie, M.: A generalized surveillance model with applications to systems safety. *IEEE Trans. Syst. Man Cybern. Pt C*. **32**, 485–492 (2002)
14. Bogdanoff, J.L., Kozin, F.: Probabilistic Models of Cumulative Damage. Wiley, New York (1985)
15. Li, W., Pham, H.: Reliability modeling of multi-state degraded systems with multi-competing failures and random shocks. *IEEE Trans. Reliab.* **54**, 297–303 (2005)
16. Feldman, R.M.: Optimal replacement with semi-Markov shock models. *J. Appl. Probab.* **13**, 108–117 (1976)
17. Ohnishi, M., Kawai, H., Mine, H.: An optimal inspection and replacement policy for a deteriorating system. *J. Appl. Probab.* **23**, 973–988 (1986)
18. Lam, C.T., Yeh, R.H.: Optimal maintenance-policies for deteriorating systems under various maintenance strategies. *IEEE Trans. Reliab.* **43**, 423–430 (1994)
19. Pham, H.: Cost optimization of a class of noncoherent systems. *Math. Comput. Model.* **15**(6), 15 (1991)
20. Dieulle, L., Berenguer, C., Gralland, A., Roussignol, M.: Sequential condition-based maintenance scheduling for a deteriorating system. *Eur. J. Oper. Res.* **150**, 451–461 (2003)
21. Li, W., Pham, H.: An inspection–maintenance model for systems with multiple competing processes. *IEEE Trans. Reliab.* **54**, 318–327 (2005)
22. Li W.: Reliability and maintenance modeling of multi-state degraded systems with multiple competing failure processes. Ph.D. thesis, Department of Industrial Systems Engineering, Rutgers University, Piscataway (2005)
23. Rardin, R.L.: Optimization in Operations Research. Prentice Hall, Piscataway (1998)
24. Wang, Y., Pham, H.: Modeling the dependent competing risks with multiple degradation processes and random shock using time-varying copulas. *IEEE Trans. Reliab.* **61**(1), 13–22 (2012)
25. Finkelstein, M.: On damage accumulation and biological aging. *J. Stat. Plan. Inference.* **139**(5), 1643–1648 (2009)
26. van Noortwijk, J.M., van der Weide, J.A.M., Kallen, M.J., Pandey, M.D.: Gamma processes and peaks-over-threshold distributions for time-dependent reliability. *Reliab. Eng. Syst. Saf.* **92**(12), 1651–1658 (2007)
27. Ye, Z.S., Tang, L.C., Xu, H.Y.: A distribution-based systems reliability model under extreme shocks and natural degradation. *IEEE Trans. Reliab.* **60**(1), 246–256 (2011)
28. Wang, Y., Pham, H.: A multi-objective optimization of imperfect preventive maintenance policy for dependent competing risk system with hidden failure. *IEEE Trans. Reliab.* **60**(4), 770–781 (2011)
29. Sari, J.K., Newby, M.J., Brombacher, A.C., Tang, L.C.: Bivariate constant stress degradation model: LED lighting system reliability estimation with two-stage modeling. *Qual. Reliab. Eng. Int.* **25**(8), 1067–1084 (2009)
30. Cossette, H., Marceau, E., Marri, F.: On the compound Poisson risk model with dependence based on a generalized Farlie-Gumbel-Morgenstern Copula. *Insurance Math. Econom.* **43**(3), 444–455 (2008)
31. Lo, S.M.S., Wilke, R.A.: A copula model for dependent competing risks. *J. R. Stat. Soc.: Ser. C: Appl. Stat.* **59**(2), 359–376 (2010)
32. Miladinovic, B., Tsokos, C.P.: Sensitivity of the Bayesian reliability estimates for the modified Gumbel failure model. *Int. J. Reliab. Qual. Saf. Eng.* **16**(4), 331–342 (2009)
33. Kaishev, V.K., Dimitrova, D.S., Haverman, S.: Modeling the joint distribution of competing risks survival times using Copula functions. *Insurance Math. Econom.* **41**(3), 339–361 (2007)
34. Bedford, T., Alkali, B.M.: Competing risks and opportunistic informative maintenance. *Proc. Inst. Mech. Eng. O: J. Risk Reliab.* **223**(4), 363–372 (2009)
35. Liu, H.: Reliability and maintenance modeling for competing risk processes with Weibull inter-arrival shocks. *Appl. Math. Model.* **71**, 194–207 (2019)
36. Fang, G., Pan, R., Hong, Y.: Copula-based reliability analysis of degrading systems with dependent failures. *Reliab. Eng. Syst. Saf.* **193**, 106618 (2020)
37. Palayangoda, L.K., Ng, H.K.T.: Semiparametric and nonparametric evaluation of first-passage distribution of bivariate degradation processes. *Reliab. Eng. Syst. Saf.* **205**, 107230 (2021)
38. Donev, V., Hoffmann, M.: Condition prediction and estimation of service life in the presence of data censoring and dependent competing risks. *Int. J. Pavement Eng.* **20**(3), 313–331 (2019)
39. Austin, P.C., Fine, J.P.: Accounting for competing risks in randomized controlled trials: a review and recommendations for improvement. *Stat. Med.* **36**(8), 1203–1209 (2017)
40. Noordzij, M., Leffondré, K., van Stralen, K.J., Zoccali, C., Dekker, F.W., Jager, K.J.: When do we need competing risks methods for survival analysis in nephrology? *Nephrol. Dial. Transplant.* **28**(11), 2670–2677 (2013)



Dr. Wenjian Li was a marketing research scientist at Javelin Direct, Inc. His research interest focuses on econometrics, forecasting, and survival analysis. Dr. Li earned his Ph.D. from Rutgers University where his primary research interests included reliability, maintenance theory, applied statistics, and manufacturing automation.

Hoang Pham is a Distinguished Professor and former Chairman of the Department of Industrial & Systems Engineering at Rutgers University. He is the author or coauthor of 7 books and has published over 200 journal articles, 100 conference papers, and edited 20 books. His numerous awards include the 2009 IEEE Reliability Society *Engineer of the Year Award*. He is a Fellow of the IEEE and IISE.



Continuous-Time Predictive Maintenance Modeling with Dynamic Decision Framework

27

Antoine Grall and Elham Mosayebi Omshi

Contents

27.1	Introduction	521
27.2	Gradual Deterioration and Predictive Maintenance	523
27.2.1	Stochastic Processes	523
27.2.2	Maintenance and Inspection Decision Rules	525
27.3	Framework of Dynamic Maintenance Policy	526
27.3.1	Grouping and Opportunistic Maintenance	527
27.3.2	Uncertainty in Degradation Models	530
27.4	Dynamic Policy with Parameter Updates	532
27.4.1	Unknown Degradation Parameters and Update	532
27.4.2	Maintenance and Inspections Decision Rule	533
27.4.3	Maintenance Cost	535
27.4.4	Decision Variables Based on Long-Term Assessment	535
27.4.5	Dynamic Policy	538
27.5	Conclusion	540
	References	540

modeling and condition-based maintenance decision rule. Then, dynamic maintenance policies are described in two different contexts: for groupings of maintenance actions and for reducing uncertainty in modeling. Finally, a particular case of dynamic preventive maintenance model is described in detail for a system with continuous degradation and unknown degradation parameters. It is based on the inverse Gaussian process with a nonperiodic inspection policy and includes parameters update.

Keywords

Predictive maintenance · Condition-based maintenance · Adaptive maintenance · Dynamic maintenance · Deterioration · Stochastic process · Inverse Gaussian · Bayesian update

Abstract

Digital technologies improve the information collected on systems and allow the development of condition-based maintenance policies and models using the remaining useful life. Accordingly, maintenance policies have evolved from a simple time-based to a more complex and competitive predictive approach. However, considering a dynamic maintenance decision framework with a self-adaptive decision rule has not been thoroughly addressed. This chapter deals with continuously deteriorating systems and focuses on dynamic maintenance policies, i.e., policies using real-time information to update the decision rule and handle the model's uncertainty. The first part presents popular stochastic processes for degradation

27.1 Introduction

Many structures or systems can be restored to working condition through maintenance actions that range from minimal repair, allowing the restart of the system, to total replacement. Depending on the equipment's operating costs or availability, it may be worthwhile to anticipate the failure and intervene while the system is still operating. Maintenance modeling addresses the overall planning of monitoring and maintenance actions throughout the equipment's useful life cycle. Possible actions can be envisaged, for example, as in [1] with an immediate online response to a detection of an unexpected sudden change in the system's operating mode. However, decision-making must be more broadly integrated into an overall supervision system.

Choosing a maintenance policy allows the maintenance decision-maker responding judiciously to questions such as "what has to be done?" and "when has it to be done?". The decision can be supported and improved by information retrieved "online."

A. Grall (✉) · E. M. Omshi
LIST3N, Université de Technologie de Troyes, Champagne, de Troyes, France
e-mail: antoine.grall@utt.fr; elham.mosayebi@utt.fr

Maintenance actions can be classified into two main categories concerning failure time. Preventive maintenance actions are programmed to prevent or delay a failure, and corrective maintenance actions occur in response to, and therefore due to, a system failure.

The development of a maintenance strategy for repairable systems requires the adoption of decision rules allowing jointly and, if possible, on the one hand, to choose the type of maintenance action to be implemented among those that are permitted, on the other hand, to plan the different actions over time. The choice and optimization of a maintenance policy require the ability to assess a given strategy's impact on the system's performance under consideration. The effects of policy need to be quantified. Performance evaluation requires developing models that represent the behavior of the maintained system and allow its characterization. All in all, it is necessary to combine:

- A representation of the system's behavior from a "new" state to failure;
- A model of the rule for monitoring and maintenance planning;
- A criterion for evaluating the performance of the maintained system.

In the last 50 years, many models have been developed and enriched under various assumptions. They allow considering different aspects of maintenance issues for systems with stochastic failure or degradation. Detailed summaries exist from different perspectives and can be found, for example, in [2–13]. Note that the models' different classifications can be proposed depending on the item which is primarily considered.

First, let us consider the system description. Based on the system modeling characteristics, it is possible to distinguish mono-component and multi-component representations. The primary representation is for single individuals or complex systems which can be considered as a whole, i.e., without describing their inner reliability structure. The next one proposes a description of the system as an assembly of parts or subsystems which can be modeled individually. For the latter, different forms of interactions between the components can be highlighted depending on the system's main characteristics under study. Consequently, different classes can be adopted. For instance, links developed for the system reliability structure (series, parallel, k out of n), or types of dependency between the components (economic, stochastic, or structural).

In parallel, the system's behavior or its components up to failure can be described in different ways. It can be based either directly on lifetimes' characterization or represent the evolution of observable conditions from "new" state to "failed" state. Direct lifetime modeling for a component

can be seen as a "two-states" representation of the system's condition ("working" and "failed"). In that case, the relevant indicator is the failure rate, which describes aging and can evolve more or less complicatedly. Moreover, modeling a system's degradation or wear evolution leads to a discrete (countable) or continuous state space with underlying stochastic processes.

Moreover, if the main classification viewpoint is guided by maintenance, models existing in the literature can be associated with the type of maintenance action (inspection, replacement, repair) and its efficiency (minimal, imperfect repair, intervention degrading the system), or with the class of decision rule (time-based rule, age-based rule, condition-based rule).

Finally, policies can be differentiated according to their assessment criteria (cost, reliability, or availability) and the considered time horizon (finite or infinite). In addition to the previous categories, one could also consider classifications depending on the methodology used to develop the cost model, whether it is based on, e.g., renewal theory or Markov decision processes. The characteristics of the information available online about the system to be maintained can also be considered.

This chapter concentrates on maintenance modeling for gradually deteriorating systems and which deterioration can be modeled continuously in time. Consequently, the degradation modeling framework will focus on continuous-time stochastic processes. The most commonly used stochastic processes in degradation modeling for prognosis and maintenance are the Wiener process, gamma process, and inverse Gaussian process. The last two processes allow describing the monotone degradation phenomenon. The gamma process has been widely considered in maintenance modeling; e.g., [14–17] to mention a few. The inverse Gaussian process has been considered more recently in this concept; see, for example, [18–20]. The Wiener process allows us to focus on the maintenance of non-monotone degradation behaviors; see, for example, [21–23], and [24].

Predictive and condition-based maintenance policies include all preventive policies for which maintenance decisions are taken based on observations of the system's current state. Decisions can possibly include future use conditions. When properly implemented, the performance of these classes of policies exceeds that of conventional static policies that do not exploit any information from online monitoring [25–27]. In this context, the main objective is to enable the appropriate use of all system monitoring information available in real-time to enhance maintenance decision-making effectiveness.

In this chapter, attention is paid mainly to the dynamic decision framework. There is no formal and precise definition of dynamic maintenance policies. It can be related to a wide range of configurations. We will consider that the dynamic maintenance decision framework refers to decision rules with

an embedded online self-modification process updated from real-time information. A dynamic maintenance policy must include updating the decision rule parameters or structure or modifying some model elements as new information becomes available. The mentioned information can be related to the system by itself (degradation level, state of other components in case of multi-component systems, ...), or some external influencing factors (covariates related to the environment, hypotheses on the system's future mission, ...). A classical condition-based maintenance policy with fixed decision variables will not be hereafter considered dynamic. Even if decisions are made based on the current observed degradation level, the decision rule by itself is set from the beginning and does not include any adaptive process.

The first part of the chapter is devoted to general definitions and notations about continuous-time degradation processes and usual maintenance decision rules. Then descriptions of the two main configurations and frameworks related to dynamic maintenance policies with gradual deterioration are given. One of the configurations corresponds to imperfect knowledge of the degradation process due to a noninformative database, identification of some random effect, or specific expert opinion. The other one is devoted to systems described as multi-component systems with possible interactions and influence of the environment.

The second part focuses on one specific case of maintenance policy that precisely illustrates the framework and adaptive process of a dynamic maintenance policy for a system described by a scalar degradation indicator. The information collected online by inspections is used to dynamically improve the maintenance decision rule and update it in real-time.

27.2 Gradual Deterioration and Predictive Maintenance

27.2.1 Stochastic Processes

In the following, the process $\{X_t\}_{t \geq 0}$ represents the evolution of the degradation indicator over time. The following is not an exhaustive presentation of the stochastic processes considered for maintenance modeling. It focuses on the ones verifying the following properties:

1. $X_0 = 0$ almost surely; i.e., the system, component or structure is considered as "new" on the date of commissioning;
2. Non-overlapping increments are statistically independent; i.e., for any partition $0 \leq t_0 < t_1 < \dots < t_n < \infty$ the increments $X_{t_1} - X_{t_0}, X_{t_2} - X_{t_1}, \dots, X_{t_n} - X_{t_{n-1}}$ represent n mutually independent random variables;

3. The increments are stationary; i.e., $\forall t, s \geq 0, X_{t+s} - X_t$ depends only on the length s of the interval, but not on its position on the time axis.

Remark that as a consequence of items 2 and 3, the distribution of $X_{t+s} - X_t$ is infinitely divisible. It means that for all integer $n > 0$ it is the sum of n independent variables with the same distribution as $X_{t+s/n} - X_t$. Therefore, appropriate infinitely divisible distributions are good candidates deriving stochastic processes. Here, we focus on continuous-state stochastic processes to model degradation. The relevance of utilizing a stochastic process for degradation modeling depends on several requirements, such as:

- Having clear physical explanations,
- Being easy to understand and use,
- Having good mathematical properties,
- Being adequate for easy incorporation of prior information and agile in dealing with covariates and random effects.

A limit value is typically considered to describe system failure in conjunction with degradation modeling. It will be referred to as L in the following. The failure time is then the first passage time of this limit. It should be noted that reaching this threshold does not necessarily correspond to a hard failure. It can be considered an exit from the acceptable use area to represent the system's inability to meet the requirements. By this definition, the system lifetime corresponds to the hitting time defined as:

$$\sigma_L = \inf\{t > 0 : X_t \geq L\}. \quad (27.1)$$

The first passage time distribution plays an essential role in reliability and maintenance. However, the remaining useful life(RUL) distribution is needed, especially in deteriorating systems. Given the current information about the system's condition at time t_k , it is related to the time duration the system can still work before failure. That means the RUL R_k of the system at t_k is defined as:

$$R_k = \inf\{r > 0 : X_{t_k+r} \geq L | X_{t_k} = x < L\}. \quad (27.2)$$

In other words, the RUL R_k is related to the first passage time σ_{L-x} by the process $\{X_t\}_{t \geq 0}$. In Eq. (27.2), x is the measured degradation level at time t_k . The important point to note here is that the information about the system is supposed to be perfect when available, which means that the system's measured degradation value is exact.

Focusing on continuous degradation modeling, three stochastic processes are widely used. Two of them correspond to nondecreasing degradation indicators and

can be easily used to model various strictly increasing degradation processes like fatigue, corrosion, and crack propagation. In contrast, the other process's distinct feature is the modeling of degradation measures, which are not necessarily monotone. The next paragraphs are devoted to short introductions of Wiener, gamma, and inverse Gaussian processes and their essential functions.

Wiener Process

The Wiener process has normally distributed increments, that is, $X_{t+s} - X_t$, for $s, t \geq 0$ has a normal distribution with drift parameter μs and diffusion parameter $\sigma^2 s$. Then, the pdf of such increment is:

$$f_{\mu s, \sigma^2 s}^W(x) = \frac{1}{\sqrt{2\pi\sigma^2 s}} \exp\left\{-\frac{(x - \mu s)^2}{2\sigma^2 s}\right\}.$$

With this consideration, the mean and variance of X_s are:

$$\mathbb{E}(X_s) = \mu s \text{ and } \mathbb{V}(X_s) = \sigma^2 s.$$

Whitmore and Seshadri [28] derived the first passage time distribution with a Heuristic method. They showed that the cumulative probability function for the first passage time T is:

$$\begin{aligned} \mathbb{P}(\sigma_L \leq s) &= \mathbb{P}\left(\max_{0 \leq t \leq s} X(t) \geq L\right) \\ &= \int_L^\infty p(x)\{1 + Q(x)\} dx \\ &= \Phi\left[\frac{\mu s - L}{\sqrt{\sigma^2 s}}\right] \\ &\quad + \exp\left(\frac{2\mu L}{\sigma^2}\right) \Phi\left[-\frac{\mu s + L}{\sqrt{\sigma^2 s}}\right]. \end{aligned} \quad (27.3)$$

Here Φ denotes the cumulative distribution function of a standard normal distribution and the functions $p(x)$ and $Q(x)$ are, respectively:

$$p(x) = (2\pi\sigma^2 s)^{-1/2} \exp\left\{-\frac{(x - \mu s)^2}{2\sigma^2 s}\right\},$$

and

$$Q(x) = \exp\{-2\mu(x - L)/\sigma^2\}.$$

Taking the derivative of (27.3) with respect to s , one obtains, after simplification,

$$f_{\sigma_L}^W(s) = \frac{L}{\sqrt{2\pi\sigma^2 s^3}} \exp\left\{-\frac{(L - \mu s)^2}{2\sigma^2 s}\right\}.$$

Reparameterizing $\lambda = L^2/\sigma^2$ and $\eta = L/\mu$, leads to the usual form of the first passage time distribution for the Wiener process. It follows an inverse Gaussian distribution. Similarly, the pdf of the RUL at time t_k can be written as:

$$f_{R_k}^W(r) = \frac{L - x_k}{\sqrt{2\pi\sigma^2 r^3}} \exp\left\{-\frac{(L - x_k - \mu r)^2}{2\sigma^2 r}\right\}.$$

Gamma Process

The gamma process is such that the degradation increment $X_{t+s} - X_t$, for $s, t \geq 0$ has a gamma distribution with shape parameter αs and scale parameter β where $\alpha, \beta > 0$. The probability density function is:

$$f_{\alpha s, \beta}^G(x) = \frac{1}{\Gamma(\alpha s)} \frac{x^{\alpha s - 1}}{\beta^{\alpha s}} \exp\left(-\frac{x}{\beta}\right) \mathbb{I}_{\{x \geq 0\}}, \quad (27.4)$$

where $\Gamma(x)$ is the gamma function, i.e.,

$$\Gamma(x) = \int_0^{+\infty} u^{x-1} e^{-u} du,$$

and $\mathbb{I}_{\{A\}}$ is equal to 1 if A is true, 0, otherwise.

The mean and variance of one increment over time s are, respectively, given by:

$$\mathbb{E}(X_s) = \alpha s \beta \text{ and } \mathbb{V}(X_s) = \alpha s \beta^2. \quad (27.5)$$

The homogeneous gamma process has been widely considered for degradation modeling, residual lifetime or reliability assessment, and maintenance modeling [17].

Let, respectively, $F_{\sigma_L}^G(s) = \mathbb{P}(\sigma_L \leq s)$ and $f_{\sigma_L}^G(s)$ denote the cumulative distribution and probability density functions of σ_L when the degradation process is a gamma process. Then:

$$\begin{aligned} \mathbb{P}(\sigma_L > s) &= \mathbb{P}(X_s < L) \\ &= \int_0^L f_{\alpha s, \beta}^G(y) dy \\ &= \int_0^L \frac{1}{\Gamma(\alpha s)} \frac{y^{\alpha s - 1}}{\beta^{\alpha s}} \exp\left(-\frac{y}{\beta}\right) dy \\ &= \frac{1}{\Gamma(\alpha s)} \int_0^{L/\beta} x^{\alpha s - 1} e^{-x} dx. \end{aligned}$$

The cumulative distribution of σ_L can be expressed as:

$$F_{\sigma_L}^G(s) = \mathbb{P}(\sigma_L \leq s) = \frac{\Gamma(\alpha s, L/\beta)}{\Gamma(\alpha s)}, \quad (27.6)$$

where $\Gamma(\alpha, x)$ is the incomplete gamma function, i.e.:

$$\Gamma(\alpha, x) = \int_x^{+\infty} u^{\alpha-1} e^{-u} du.$$

Then, the corresponding pdf is :

$$f_{\sigma_L}^G(s) = \frac{\alpha}{\Gamma(\alpha s)} \int_{L/\beta}^{+\infty} \left(\log(x) - \frac{\Gamma'(\alpha s)}{\Gamma(\alpha s)} \right) x^{\alpha s-1} e^{-x} dx,$$

which is difficult to compute in practice. Usually, a simpler approximate distribution is used for computation. The approximated pdf of the first passage time for the gamma process is [29]:

$$f_{\sigma_L}^G(s) \approx \frac{L}{\beta \sqrt{2\pi \alpha s^3}} \exp \left\{ -\frac{\alpha \left(s - \frac{L}{\alpha \beta} \right)^2}{2s} \right\}.$$

Therefore, the pdf of RUL at time t_k can simply be approximated by:

$$f_{R_k}^G(r) \approx \frac{L - x_k}{\beta \sqrt{2\pi \alpha r^3}} \exp \left\{ -\frac{\alpha \left(r - \frac{L - x_k}{\alpha \beta} \right)^2}{2r} \right\}.$$

Inverse-Gaussian Process

According to this process, the degradation increment $X_{t+s} - X_t$ for $s > 0$, $t \geq 0$ follows an inverse Gaussian distribution (or Wald distribution) with mean parameter s/δ and shape parameter λs^2 . The related pdf and cdf are defined for $x > 0$ as follows:

$$f_{s/\delta, \lambda s^2}^{IG}(x) = \sqrt{\frac{\lambda s^2}{2\pi x^3}} \exp \left\{ -\frac{\lambda}{2x} (\delta x - s)^2 \right\}$$

and

$$F_{s/\delta, \lambda s^2}^{IG}(x) = \Phi \left(\sqrt{\frac{\lambda}{x}} (s - \delta x) \right) - \exp(2s\delta\lambda) \Phi \left(-\sqrt{\frac{\lambda}{x}} (s + \delta x) \right),$$

where $\delta, \lambda > 0$ and Φ is the cdf of the standard normal distribution. The mean and variance of the increment over time s are:

$$\mathbb{E}(X_s) = \frac{s}{\delta} \text{ and } \mathbb{V}(X_s) = \frac{s}{\delta^3 \lambda}. \quad (27.7)$$

The inverse Gaussian process has been considered more recently for degradation modeling [30, 31].

The cumulative distribution function of σ_L when the the degradation process is an inverse Gaussian process can be written as:

$$F_{\sigma_L}^{IG}(s) = \mathbb{P}(\sigma_L \leq s) = \Phi \left(\sqrt{\frac{\lambda}{L}} (s - \delta L) \right) - \exp(2s\delta\lambda) \Phi \left(-\sqrt{\frac{\lambda}{L}} (s + \delta L) \right).$$

By derivation with respect to s , the pdf is:

$$f_{\sigma_L}^{IG}(s) = 2\sqrt{\frac{\lambda}{L}} \Phi \left(\sqrt{\frac{\lambda}{L}} (s - \delta L) \right) - 2\delta\lambda \exp(2s\delta\lambda) \Phi \left(-\sqrt{\frac{\lambda}{L}} (s + \delta L) \right).$$

Accordingly, the RUL has the following cdf:

$$F_{R_k}^{IG}(r) = \Phi \left(\sqrt{\frac{\lambda}{L - x_k}} (r - \delta(L - x_k)) \right) - \exp(2r\delta\lambda) \Phi \left(-\sqrt{\frac{\lambda}{L - x_k}} (r + \delta(L - x_k)) \right). \quad (27.8)$$

27

27.2.2 Maintenance and Inspection Decision Rules

It is necessary to choose a specific structure of decision rule for continuous deterioration models. Most the existing works propose parametric structures based on control limit decision rules. The aim is to decide whether a maintenance action must be performed or not and define the next inspection time. All such decisions must be made given the current measured degradation level.

For this aim, let $\{T_n\}_{n \in \mathbb{N}}$ with $T_0 = 0$, denote the sequence of inspection times such that:

$$T_{n+1} = T_n + \Delta T_n.$$

Inspections can be planned either periodically or aperiodically. In the periodic inspection planning, the system is inspected at each T unit of time. That means $\Delta T_n = T$, for all n . In this case, the constant T is a decision variable and must be optimized. Periodic inspection planning is easy to implement in practice, but some concerns may arise, especially in deteriorating systems. As a matter of fact, the probability of failure increases with time or usage as the degradation level increases. Consequently, when a system is new, there is no need to have numerous inspections; as it ages, more frequent inspections are needed. Hence, in some cases, the maintenance cost will be high in the case of periodic inspections.

Aperiodic inspection planning is related to the current state of the system. It is self-adaptive and takes into account the most recent information about the system state. This alternative has become more popular for gradually deteriorating systems, see for example [21, 32–34]. Nonperiodic inspection scheduling leads to better results than periodic one as the latter can often be considered a limit case of aperiodic planning.

To implement aperiodic inspection planning, a decreasing function of system state, m , can be used to set the time between two inspections, i.e.,

$$\Delta T_n = m(X_{T_n}).$$

The general idea is to decrease the time between successive inspections as the degradation level increases. References [32, 35] used a special linear case of such function. Characteristics of this linear function were considered as decision variables to optimize. Utilizing the remaining useful life function is also another appropriate option [36, 37]. In this case:

$$\Delta T_n = \tau_p(X_{T_n}),$$

where

$$\tau_p(X_{T_n}) = \{\Delta t : \mathbb{P}(X_{T_n+\Delta t} \geq L | X_{T_n}) = p\}. \quad (27.9)$$

In other words, $\tau_p(X_{T_n})$ is the p -quantile of the remaining useful life distribution. Here, p is a decision variable. It is worthy to note that RUL-based inspection planning provides a reliability (safety) level equal to $(1 - p)$.

Along with inspection scheduling, maintenance actions must be arranged. To this aim, fixed thresholds are considered on the degradation level for preventive repairs or replacements. That means the systems with degradation higher than a specified level are highly risky, hence better to be replaced preventively. Replacements can alternatively be triggered as soon as the interval between two successive inspections is less than a given threshold. It is also reasonable because, as already mentioned, only the systems with a higher chance of failure need frequent inspections. In all these cases, the thresholds are decision variables that have to be optimized.

For more details about maintenance actions based on a threshold of degradation level, let $X_{T_n^-}$ ($n \in \mathbb{N}$) refer to the degradation level at the corresponding inspection time before potential maintenance action. Classically, with a preventive threshold M , the structure of the decision rule for perfect maintenance is as follows:

- If $X_{T_n^-} \geq L$, the system has failed and is correctively replaced. The next inspection is planed at time $T_n^- + m(0)$.

- If $M \leq X_{T_n^-} < L$, the system has not yet failed but is considered as “close to failure.” As a consequence, preventive action is taken immediately. The next inspection is planed at time $T_n^- + m(0)$.
- If $X_{T_n^-} < M$, the system is still properly functioning, and there is no need for replacement. The system is left as it is. The next inspection is planed at time $T_n^- + m(X_{T_n^-})$.

In this consideration, $m(\cdot)$ can equivalently be replaced by $\tau_p(\cdot)$. The decision-maker must determine the decision variables to optimize a given assessment criterion, usually the global maintenance cost. Decision variables are typically M and parameters of the function m or τ_p . In the case of discrete degradation modeling, i.e., with discrete state space, it may be possible to find the optimal decision structure, e.g., with Markov decision theory.

Note that, in imperfect maintenance actions, the structure of the decision rule is not so straightforward and requires additional elements and parameters. For instance, selection between imperfect preventive repair and perfect preventive replacement can be based on two preventive thresholds M_1 and M_2 with $0 < M_1 < M_2 < L$. Imperfect action takes place as the system state at inspection time is between M_1 and M_2 . Perfect ones are needed when the system state is higher than M_2 and less than L ; see, for example, [38]. Describing the gain of imperfect intervention with a specified distribution and defining appropriate characteristics for this distribution is also done in [39].

27.3 Framework of Dynamic Maintenance Policy

A dynamic maintenance policy is a maintenance policy that implements online modifications of an initial inspection/maintenance planning from new monitoring information. It adapts itself to the level of information which is available at a given time. The framework of dynamic maintenance covers a vast range of configurations. New information can be related directly to the maintained system, its environment or future usage. Two different cases are considered hereafter, which can lead to the development of dynamic maintenance policies. The first one focuses on multi-unit or multicomponent systems with dependencies between components. In that case, the dynamic characteristic is directly related to the grouping of maintenance actions, which requires successive updates for future action planning. The terminology “dynamic maintenance” has initially been introduced in this context. The second case deals with maintenance problems when the degradation phenomenon is not fully characterized. More precisely, a degradation model is chosen, but its parameters are unknown. In that case, successive inspections have to be considered online to

improve the maintenance model dynamically in real time. These two cases are described more precisely in the next two sections.

27.3.1 Grouping and Opportunistic Maintenance

Complex deteriorating systems or subsystems can be considered as a whole and characterized by a unique degradation indicator. Nevertheless, such systems may be composed of components, some of which deteriorate individually. The interaction between these components and their related degradation leads to the system's overall deterioration. Different specific information is collected if these components can be inspected and monitored separately. Therefore, it is interesting to consider a multivariate model that describes possible dependencies between components or the effect of specific missions on certain parts of the system. A dynamic decision framework can improve maintenance decision-making for such multi-component systems or fleets of units, especially when the timing of interventions needs to be managed rationally.

This section focuses on how dynamic maintenance can be implemented for multicomponent systems to handle short-term information obtained online. It is interesting for such systems to take advantage of dependencies between components, leading to maintenance opportunities. Different classifications of dependencies can be found in the literature. From the extended scheme proposed in [40], four different types of dependencies exist in contributions on condition-based maintenance policies: structural dependency, stochastic dependency, resource dependency, and economic dependency. Briefly, structural dependency can be related to the system reliability block diagram or some technical dependence when the maintenance of one component requires replacing other components. Stochastic dependency describes the relationship between the degradation of different components due to, e.g., load sharing, common causes of failure, or when the failure of one component modifies others' deterioration. Resource dependency deals with restrictions due to limited resources for maintenance, whether in spare parts, logistics, tools, maintenance workers, or budget. Finally, the system is subject to negative or positive economic dependencies when maintaining several components together is respectively more expensive or less expensive than maintaining them separately.

Dynamic maintenance grouping strategies have been developed mainly for economic dependencies in the context of continuously deteriorating multicomponent systems. More precisely, most of the models consider setup costs as follows. When maintenance is triggered, a system-dependent cost or setup cost S is incurred systematically. It is counted

only once, regardless of the number of components to be maintained simultaneously. In addition, let c_i be the individual maintenance cost of component i . Hence if all the components from the subset of components E_m are maintained simultaneously, it incurs the total maintenance cost $S + \sum_{i \in E_m} c_i$ at that time. Nonsimultaneous maintenance actions would lead to a cost equal to $\sum_{i \in E_m} (S + c_i)$. Due to the possibility of grouping maintenance activities, relevant groupings can reduce maintenance costs over a given planning horizon. The failure of a critical component can be considered an opportunity for additional preventive maintenance actions. Short-term additional information can be obtained in real time. For example, it can be linked to changes in the system's operational environment. This type of information is interesting to dynamically improve groupings on a rolling short-term horizon [41].

The seminal work on dynamic maintenance policy for grouping maintenance is developed in [42], which deals with the lifetime modeling of components. It does not consider continuously deteriorating systems but allows us to define the "dynamic" characteristic of maintenance policy. This characteristic is related to the availability of information and the choice to take it into account "online" with possible modifications of the decision rule's structure or decision variables' updates. Specifically, the authors in [42] consider an iterative procedure that characterizes dynamic rules as described in Fig. 27.1.

First, long-term tentative maintenance planning is defined, which consists of individual maintenance planning for each activity or component. It is based on long-run maintenance costs and stationary states. For each component, a penalty cost function is jointly derived to assess the drawback due to shifting the execution time of maintenance actions from the tentative planning. At a given time t , current information about the system state and the planned future solicitation is introduced. That step is considered on a short-term time horizon and allows us to shift the tentatively planned times obtained from the first step. Based on the penalty cost function, an optimal grouping structure is derived on the short-term horizon and iteratively updated when it is needed. The reiteration of this procedure at different successive times corresponds to the "rolling-horizon step." As already mentioned, the work in [42] does not consider gradual deterioration but rather lifetime modeling. It is devoted to series systems, and an algorithm based on dynamic programming technics is proposed for optimal grouping. The information about the system state available online is related to replacement and repair times of components and their usage rates. In [42], the latter can be updated. It illustrates the possibilities of their approach. Short-term knowledge about the future tasks planned for the system is also included when it is available. The "dynamic" property of the maintenance decision rule is related explicitly to the maintenance groupings' successive

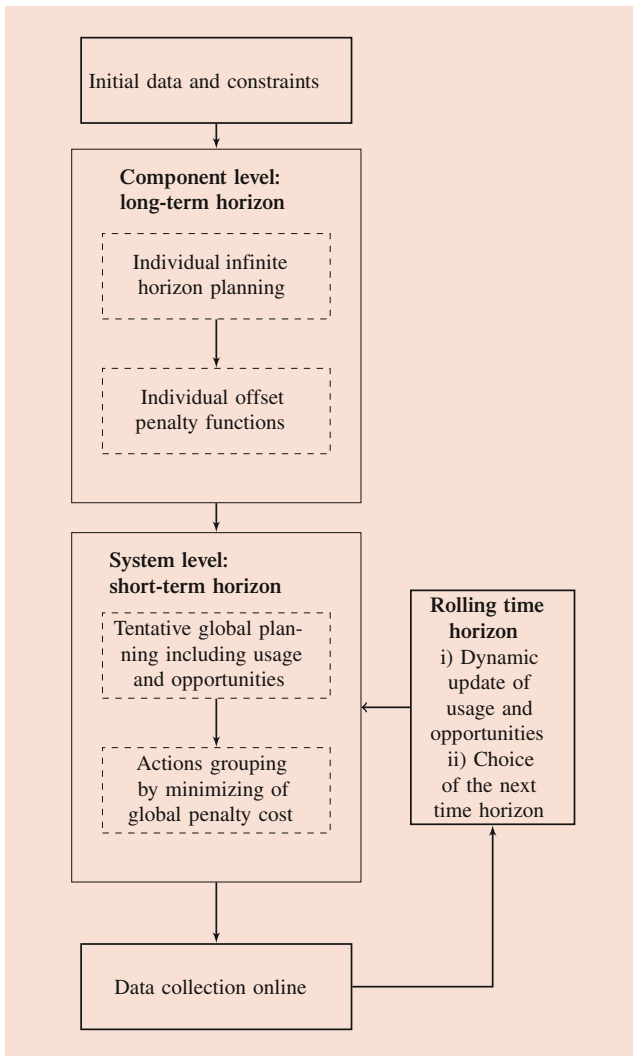


Fig. 27.1 The dynamic process for multiunit systems

updates. Other work has been subsequently published following the same idea; we refer to [43] to see a more complex structure for multicomponent systems. It is worthy to note that the optimal grouping strategy can be obtained via a Genetic algorithm.

Let's now consider systems that gradually deteriorate with time or usage. In such cases, information about the degradation level can be obtained by inspections or continuous monitoring. Condition-based or predictive maintenance policies can be implemented, and dynamic procedures may be derived. To illustrate and give more details about these dynamic policies, we focus hereafter on the works of Bouvard et al. [44], and Van Horenbeek and Pintelon [45]. It highlights the main characteristics of the dynamic maintenance framework for multicomponent deteriorating systems.

Consider a system with n components that deteriorate gradually. Each component is characterized by its degradation indicator $X_i(t)$ obtained by synthesizing

information from several sensors or directly representing a physical variable. In the mentioned papers, [44] and [45], the degradation processes are modeled classically with homogeneous gamma processes described in Sect. 27.2.1. Information about the degradation level of each component can be obtained at inspection times. Component i is considered failed when its degradation level crosses a failure threshold L_i , which may be a constant as in [44] or a random variable depending on operating load, operating environment, etc, as in [45]. In that case, information about the value of L_i can be obtained on a given time horizon. Each component can be preventively or correctively maintained, and the structure of the costs is as described previously: the system-dependent or set-up cost S is incurred once at each maintenance time. With the same notations as in [44, 45], the total maintenance cost on a finite planning horizon including m maintenance times referred as t_i , $i = 1, \dots, m$ and ending at time t_m can be written in a very general way as:

$$C(t_m) = mS + \sum_{i=1}^m \sum_{j \in G_i} (c_{j,p} \mathbb{I}_{X_j(t_i) \leq L_j} + c_{j,c} \mathbb{I}_{X_j(t_i) > L_j}),$$

where G_i is a subset of $\{1, \dots, n\}$, which refers to the numbers of the components maintained at time t_i . $c_{i,p}$ and $c_{i,c}$ are, respectively, preventive and corrective maintenance cost for component i . \mathbb{I}_A is equal to 1 if A is true, 0 otherwise. The main issue is then determining the groups G_i and the dates of intervention on the considered period to minimize $C(t_m)$.

The dynamic maintenance policy aims to optimally define the maintenance times and determine the associated groups of components to be maintained simultaneously. The proposed algorithms in [44, 45] follow the same procedure as in [42]. Each component is firstly considered individually. A provisional individual schedule is derived based on a classical maintenance policy for single unit systems, and it is optimized in order to minimize the long-run cost rate. This cost is evaluated by taking into account updated monitoring information. If new information becomes available and if necessary, the long-run cost rate expression can be updated.

Advanced continuous condition-based or predictive maintenance policies for single-unit systems could be considered for individual tentative planning. However, these decision rules use the last observed level of degradation to determine online the next time between inspections and the decision about the maintenance date. This real-time adaptive process makes the aggregation step difficult. Consequently, an age-based maintenance policy is chosen by the authors in [44] and [45] for the sake of simplicity. According to such a policy and considering each component individually, the component i is replaced at failure time or age θ_i^* [7]. In this case, the long-run cost rate can be expressed analytically from

the conditional reliability of components. The associated asymptotic cost rate can be obtained from renewal theory for component i as:

$$\begin{aligned} C_i^\infty(\theta) &= \lim_{t \rightarrow \infty} \frac{C_i(t, \theta)}{t} = \frac{E[C_i(T_i^r, \theta)]}{E[T_i^r]} \\ &= \frac{c_{i,p} + (c_{i,c} - c_{i,p})F_i(\theta)}{\int_0^\theta (1 - F_i(u))du}, \end{aligned} \quad (27.10)$$

where $C_i(t, \theta)$ is the cumulative maintenance cost up to time t and T_i^r represents the first replacement time, i.e., the length of the first renewal cycle. Moreover, F_i is the cumulative distribution function of the failure time of component i denoted by σ_{L_i} . That means:

$$F_i(\theta) = \mathbb{P}(\sigma_{L_i} \leq \theta) = \mathbb{P}(X_i(\theta) \geq L_i).$$

With notations from Sect. 27.2.1, it can be rewritten:

$$F_i(\theta) = G_{\sigma_{L_i}}^G(\theta) = \frac{\Gamma(\alpha_i \theta, L_i / \beta)}{\Gamma(\alpha_i \theta)}.$$

Aforementioned, the policy's optimal decision variable is the limit age of replacement θ_i^* , which minimizes $C_i^\infty(\theta)$. Authors in [44] and [45] made calculations with a discretized degradation process in order to assess such optimal decision variables. The cumulative distribution function of the failure time for the initial gamma process is converted into a discretized reliability function that is easier to handle numerically. Let the step of this discretization be hereafter denoted as δt . According to this discretization, $t_k = k\delta t$ and the conditional reliability over one time step δt given that the component's current age is t_k is:

$$\begin{aligned} R_{i,k} &= R(t_{k+1} | \sigma_{L_i} > t_k) \\ &= \mathbb{P}(\sigma_{L_i} > t_k + \delta t | \sigma_{L_i} > t_k), \end{aligned}$$

and by definition, it comes:

$$\begin{aligned} R_{i,k} &= \frac{\mathbb{P}(\sigma_{L_i} > t_{k+1})}{\mathbb{P}(\sigma_{L_i} > t_k)} \\ &= \frac{1 - F_i(t_{k+1})}{1 - F_i(t_k)}. \end{aligned}$$

The values of $R_{i,k}$ can then be obtained as a discretization of the failure probability distribution $F_i(\cdot)$, which is obtained by Monte-Carlo simulation in [45] (with random failure limit L_i). Then, the maintenance cost rate can be rewritten as:

$$C_i^\infty(\theta) \simeq \frac{c_{i,p} + (c_{i,c} - c_{i,p}) \left(1 - \prod_{k=0}^{l-1} R_{i,k}\right)}{1 + \sum_{p=2}^l \prod_{k=0}^{p-2} R_{i,k}}, \quad (27.11)$$

with $\theta = l\delta t$. Maintenance downtimes can also be introduced additionally in the previous expression.

Note that up-to-date information, i.e., the current degradation level of the component, only influences short-term predictions and planning. On a long-term basis, the stationary state or mean behavior is considered. Nevertheless, it is possible to consider the last measured degradation level of the component jointly with the expression of the long-run cost rate. Let the observed degradation level of component i at time t_0 be denoted by $X_i(t_0) = x_{i,0}$. It is possible to consider the cost expression from Eqs. (27.10) or (27.11) by replacing L_i with $L_i - x_{i,0}$ in the expressions above. It is equivalent to considering cycles where the initial state is $x_{i,0}$ instead of 0. Let $\theta_{i,0}^*$ be the value of θ , which minimizes this cost. The next preventive replacement given the degradation level $x_{i,0}$ at time t_0 can be scheduled at time $t_0 + \theta_{i,0}^*$.

As an illustration, let's focus on the expression of cost in continuous case specified by Eq. (27.10). Given the current degradation level of the component i at time t , the maintenance cost function can be written as a function of the conditional cumulative distribution function F_i , i.e.:

$$C_i^\infty(\theta | X_i(t) = x_{i,t}) = g(F_i(\theta | X_i(t) = x_{i,t})),$$

where g is a known function depending on maintenance unit costs and:

$$\begin{aligned} F_i(\theta | X_i(t) = x_{i,t}) &= 1 - R_{i,t}(\theta | X_i(t) = x_{i,t}) \\ &= \mathbb{P}(X_i(t + \theta) \geq L | X_i(t) = x_{i,t}), \end{aligned}$$

where $R_{i,t}(\theta | X_i(t) = x_{i,t})$ is the conditional reliability of component i at time θ with last inspection at time t . It can be obtained from the expression of the hitting time, as defined in Sect. 27.2.1. At time t , two different versions of the cost are considered as criteria to be minimized for the determination of the age of preventive maintenance:

- the criterium of short-term optimal maintenance time for component i takes into account the last monitoring information. It means that at inspection time t , this criterium is the cost:

$$C_i^{ST}(\tau) = C_i^\infty(\tau | X_i(t) = x_{i,t}).$$

It can also include short-term prognosis information about future missions;

- the criterium of long-term optimal maintenance time for component i is based on the maintained component's stationary behavior. It is obtained considering the component is new at time t , i.e., the cost is

$$C_i^{LT}(\tau) = C_i^\infty(\tau | X_i(t) = 0).$$

In the case of imperfect maintenance, the expectation over the degradation level of component i just after maintenance is considered. It requires the knowledge of the corresponding degradation level distribution.

The value $\tau_{i,1}^*$ which minimizes $C_i^{ST}(\tau)$ gives the time interval before the next tentative maintenance time. For the following maintenance times the value $\tau_{i,+}^*$ which minimizes $C_i^{LT}(\tau)$ is considered. In other words, the k -th planned maintenance time for component i takes place at time $t + \tau_{i,1} + (k-1)\tau_{i,+}$ in the absence of failure. Downtime due to maintenance duration can be considered but is neglected here. In both cases, the cost expression comes from a formulation over an infinite horizon and uses renewal properties. The dynamic decision-making process leads to using it only on a limited time horizon and then updating it.

A strategy for maintenance groupings must be defined to exploit economic dependencies represented by setup costs. For this purpose, a penalty cost function is defined along with the initial individual tentative planning. It allows evaluation of how the cost is affected by shifting maintenance activities from the previous tentative planning for each component. Two cases are considered by including or not the knowledge about $X_i(t)$, in accordance with the way short-term and long-term optimal maintenance times have been determined previously. In case of failure of component i at time t , the penalty function is infinite if the replacement is delayed and null if it is moved forward. Let G_j denote a group of components to be maintained at the same time. Assume the maintenance of the group takes place at time t_g . The penalty cost associated with this group is denoted by $H_{G_j}(t_g)$, which is the sum of short-term penalty costs of all components from G_j . The optimal maintenance time $t_{G_j}^*$ for group G_j is the one minimizing H_{G_j} . When the maintenance time is determined for the group, it is possible to evaluate the savings obtained if the maintenance of all the elements of the group G_j is simultaneously executed at time $t_{G_j}^*$. It is given by:

$$Q_{G_j}(t_{G_j}^*) = (|G_j| - 1)S - H_{G_j}(t_{G_j}^*),$$

where $|G_j|$ is the number of elements in G_j and S is the setup cost.

Maintenance groupings are considered on a given planning horizon HP, i.e., considering all the tentative planned maintenance actions within this time horizon. The length of the planning horizon can be determined in different ways. For example, it is defined as:

- HP = $\max_{i \in \{1, \dots, n\}} \{\tau_{i,1}\}$ in [44];
- HP = $\max_{i \in \{1, \dots, n\}} \{\tau_{i,1} + \tau_{i,+}; \epsilon_i\}$ where ϵ_i is the time between two successive inspections of component i in [45].

Information about production planning or intended short-term use could be considered. The planning horizon is chosen to include maintenance actions on all the components and has to be short enough to allow optimal groupings. The set of optimal groups G_j over horizon HP is determined heuristically based on the grouping algorithm from [42]. According to the dynamic framework, a rolling-horizon approach is considered. The planning horizon is modified whenever new monitoring information is obtained, and the groups are updated accordingly. In summary, the problem complexity in grouping strategies is mainly due to the number of components and the juxtaposition of multiple degradation processes. The degradation models and individual inspection and maintenance policies are chosen as simply as possible. They are supposed to be perfectly known. The next section deals with another difficulty, not related to the system structure but to the degradation model itself. The different case of a single subsystem or component is considered but with a degradation model which is not entirely known. It leads to additional uncertainty, which can be handled with the help of monitoring information.

27.3.2 Uncertainty in Degradation Models

The problem of determining the optimal maintenance strategy for deteriorating systems has been studied extensively. Almost all of these studies assume that the degradation process and its parameters are known with certainty, although this is usually not the case in practice. Based on the situation, uncertainty can be considered in the model in different ways. A common approach is to assume a specific degradation process and consider the related parameters as unknown [46, 47]. This assumption is helpful when a flexible degradation process is chosen, which can be appropriate in various cases.

With the consideration mentioned earlier, the parameter estimation must be regarded. The parameters can be estimated either by gathering data and then using the frequentist perspective or using the Bayesian approach. In the latter case, the expert judgment or the previous information about the system can be considered for prior distribution. The more information is in hand; the more informative the prior distribution will be. Otherwise, noninformative priors can be assumed. Here, we refer to some examples of such estimations. Paroissin [48] provided recursive estimators based on the moments of the gamma process. As the author mentioned, this online estimation method could be applied jointly with a condition-based maintenance policy. Kallen and van Noortwijk [46] employed the Bayes method to estimate the unknown degradation parameters. Iterative Bayesian updates of degradation parameters can be found in [49–51]. Liu et al. [52] considered a maintenance model taking account of both

degradation and aging. Their model updates the degradation processes' parameters at each inspection with maximum a posteriori (MAP) estimation when a new observation is in hand.

Attempts to analyze the influence of Bayesian updates on maintenance decision-making have been made in [47,53,54]. Flage et al. [53] gave a general description of aperiodic condition-based maintenance by focusing on safety constraints. Papers [47,54] also deal with a dynamic and adaptive predictive maintenance policy for degradation processes with unknown parameters. The authors introduced a heuristic way to assess the long-run maintenance cost in [47], while [54] includes the simulation-based version of the research. Adaptive preventive maintenance policies in lifetime modeling with Bayes' theorem can also be found in [55,56].

The uncertainty in the parameters can also happen due to a sudden change in the environment. Fouladirad and Grall [57] and Fouladirad et al. [58] studied the case in which the degradation rate is subject to sudden changes. Fouladirad and Grall [59,60] investigated the situation that after such an abrupt change in the system degradation rate, the new parameters are unknown.

The change in degradation rate can also be related to imperfect actions for a system as preventive maintenance. Generally, two main effects have been considered for imperfect maintenance actions: (i) restoring the system to a state between good-as-new and bad-as-old and (ii) changing, or in severe cases, accelerating the degradation rate. Therefore, the degradation rate after imperfect maintenance action may be different and unknown. The second effect, which usually is referred to as imperfect maintenance's negative effect, must be regarded carefully. To deal with this problem, Zhang et al. [61] employed the random improvement factor model when a Wiener process governs the underlying degradation. They considered that the degradation rate after a repair would be $bv(t)$ with $0 < b < 1$, while the degradation rate before the repair is $v(t)$. The random variable b is called the degradation-rate-reduction factor and has a specific distribution. Considering this, each imperfect maintenance action has a different degree of impact on the rate of deterioration. They used the quasi-Monte Carlo method to estimate the fixed model parameters. Then the impact of each maintenance action is dynamically estimated with filtering techniques. Zhao et al. [62] also employed the random improvement factor to model the system's condition after an imperfect repair. Consider that an imperfect repair takes place at time T_i which is driven by the remaining useful lifetime information, then:

$$X_i^+ = a_i X_i,$$

where $0 < a_i < 1$, and X_i, X_i^+ are, respectively, the condition of system before and after the maintenance action at time T_i . Another work related to this issue is Shen et al. [63].

They considered that after j -th imperfect maintenance action the threshold of failure L will be substituted by $(1 - \delta^j)L$ where $0 < \delta < 1$. The reason for such a description is that after an imperfect maintenance action, the condition of a system will not be better than its condition at previous maintenance. Hence, the system will fail sooner. To mention other works in which the system's state gets worse by increasing the number of imperfect maintenance, we may refer to Huynh [64,65]. Do et al. [36] and Chen et al. [39] considered the gamma process and tried to model the negative effect of imperfect maintenance by defining additive random models for the rate of the degradation procedure. Assume that the shape parameter of degradation process before and after j -th imperfect maintenance are, respectively, v_{j-1} and v_j . To describe the effect of j -th imperfect maintenance on the system deterioration speed, Do et al. [36] suggested

$$v_j = v_{j-1} + \varepsilon_j,$$

where ε_j follows an exponential distribution. While, Chen et al. [39] proposed:

$$v_j = \rho v_{j-1} + \text{error},$$

where $\rho > 1$ and the error term represents the effect of the difference between the environment before and after the maintenance action.

Another source of uncertainty may be due to heterogeneity. Many papers assume that the parameters are fixed for all units across the population in maintenance or degradation modeling. This assumption is not usually correct since units' degradation characteristics can be identical due to different environments and usage. Utilizing a random effect model can help take account of heterogeneities commonly observed among a product population. As an example of such consideration in maintenance policies, we can refer to Chen et al. [18]. They have chosen an inverse Gaussian process for modeling degradation and assumed the inverse of the degradation rate has a truncated normal distribution. Hence different products may deteriorate at various speeds. They also supposed the chosen distribution for random effect can be updated when more degradation observations are available.

The dynamic maintenance framework is of great interest in the case of uncertainty in degradation models and the grouping of maintenance actions. It allows taking advantage of real-time monitoring data to improve decision-making according to the general framework described above. The following section will detail more precisely an example of a dynamic maintenance framework introduced in [47]. It is specifically developed for a deteriorating system with unknown deterioration parameters. This system can be monitored only by inspections. The maintenance decision rule

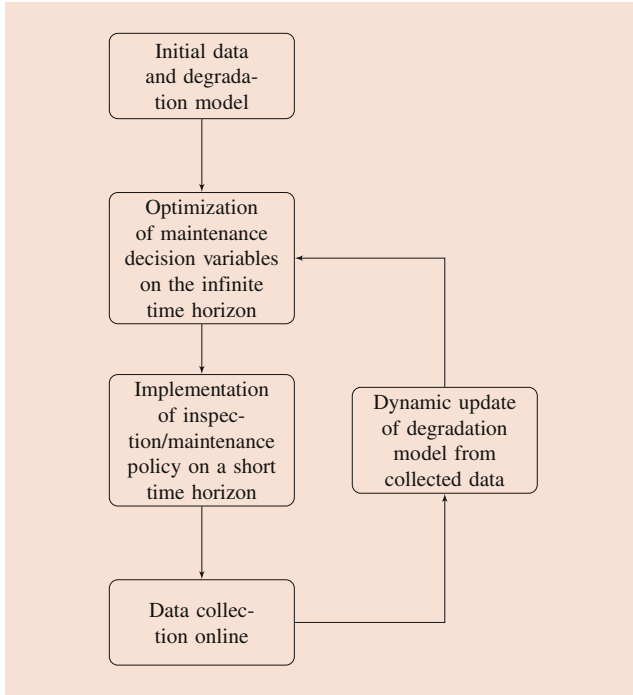


Fig. 27.2 The dynamic process of the model update from monitoring data

belongs to the class of condition-based predictive policies, which allows the optimization of aperiodic inspection times as well as system replacements or perfect repairs. According to the dynamic process described in Sect. 27.3.1, the general scheme is based on alternating long-run cost rate optimization for a given planning horizon and successive updates using the information collected online as described in Fig. 27.2.

27.4 Dynamic Policy with Parameter Updates

27.4.1 Unknown Degradation Parameters and Update

Degradation Process

As it is already mentioned in Sect. 27.3.2, the assumption of having a specified degradation process with an unknown parameter is appropriate when the degradation process is flexible enough to fit a various range of degradation data. As Wang and Xu [30] showed, the inverse Gaussian (IG) process is a suitable model to handle such a situation. On the other hand, the IG process has a close relationship with normal distribution; hence Bayesian inference can be convenient. For these reasons, this section will focus on the IG process while describing dynamic condition-based maintenance with Bayesian updates. More precisely, the system degradation

behavior will be modeled by an IG process, as described in Sect. 27.2.1.

If degradation parameters are unknown, expert opinion and other information about degradation can be used to choose a prior distribution for parameters. This prior distribution can be updated in the Bayesian framework when new information arises. For the sake of simplicity and the ease of calculations, conjugate priors will be considered preferably. With the same notations as in Sect. 27.2.1, two unknown parameters of the IG process are respectively denoted by δ and λ . Focusing on conjugate priors, the joint prior distribution of (δ, λ) is given by $f(\delta, \lambda) = f_1(\delta|\lambda)f_2(\lambda)$ where λ is gamma distributed with shape and scale parameters α and β such that $\mathbb{E}[\lambda] = \alpha\beta$ and $\mathbb{V}[\lambda] = \alpha\beta^2$, i.e.,

$$f_2(\lambda) = \frac{\lambda^{\alpha-1}}{\Gamma(\alpha)\beta^\alpha} \exp\{-\lambda/\beta\}, \quad (27.12)$$

and given λ , δ is normally distributed with mean ξ and variance σ^2/λ , i.e.:

$$f_1(\delta|\lambda) = \sqrt{\frac{\lambda}{2\pi\sigma^2}} \exp\left\{-\frac{\lambda(\delta - \xi)^2}{2\sigma^2}\right\}. \quad (27.13)$$

The hyper-parameters of the degradation model, α , β , ξ and σ , can be updated with new information.

Remaining Useful Lifetime

In the rest of this chapter, when the inspection times and associated degradation level are noted, respectively, t_k and x_k , the notation $R_k = R_{t_k}$ will be used for the RUL R_{t_k} as defined in (27.2).

As mentioned in Sect. 27.2.1, a failure occurs once degradation exceeds the threshold L . The system lifetime σ_L and the RUL R_k are defined, respectively, by Eqs. (27.1) and (27.2). RUL's cdf is given by:

$$F_{R_k}(r|X_t = x_t) = \int_0^\infty \int_{-\infty}^{+\infty} F_{R|\delta,\lambda}^{IG}(r|\delta, \lambda, x_t) f_1(\delta|\lambda) f_2(\lambda) d\delta d\lambda,$$

where $F_{R|\delta,\lambda}^{IG}(r|\delta, \lambda, x_t)$ is RUL cdf in case of IG degradation process with known parameters, as given in Eq. (27.8). Reference [66] proposed developments leading to:

$$\begin{aligned} F_{R_k}(r|X_t = x_k) &= 1 - \frac{\sqrt{\beta}}{2\pi} \frac{\Gamma(\alpha + 1/2)r}{\Gamma(\alpha)} \\ &\times \int_0^{L-x_t} z^{-3/2} (\sigma^2 z + 1)^{-1/2} \\ &\times \left(1 + \frac{\beta(\xi z - r)^2}{2z(\sigma^2 z + 1)}\right)^{-(\alpha+1/2)} dz. \end{aligned} \quad (27.14)$$

which can also be rewritten as a function of the standard t -distribution.

Bayesian Update

Consider that the system degradation level has been measured at $k+1$ successive times $t_0, \dots, t_k, k \in \mathbb{N}$. The observed data are

$$\text{Data} = \{(t_0, x_0), (t_1, x_1), \dots, (t_k, x_k)\}.$$

Let $\Delta x_i = x_i - x_{i-1}$ and $\Delta t_i = t_i - t_{i-1}$ denote the observed degradation increments and the corresponding time intervals, for $i \in \{1, \dots, k\}$. Then, the likelihood function of degradation parameters, δ and λ , is:

$$L(\delta, \lambda | \text{Data}) \propto \lambda^{\frac{k}{2}} \exp \left\{ -\frac{\lambda}{2} \sum_{i=1}^k \frac{(\Delta x_i \delta - \Delta t_i)^2}{\Delta x_i} \right\}, \quad (27.15)$$

and given the observed data, the joint posterior distribution of (δ, λ) is such that:

$$f(\delta, \lambda | \text{Data}) \propto L(\delta, \lambda | \text{Data}) f_1(\delta | \lambda) f_2(\lambda).$$

From Eqs. (27.12), (27.13) and (27.15) it can be shown that (see for example [47]):

$$f(\delta, \lambda | \text{Data}) = f_1(\delta | \lambda, \text{Data}) f_2(\lambda | \text{Data}),$$

where

$$f_1(\delta | \lambda, \text{Data}) = \sqrt{\frac{\lambda}{2\pi\sigma^{*2}}} \exp \left\{ -\frac{\lambda(\delta - \xi^*)^2}{2\sigma^{*2}} \right\},$$

$$f_2(\lambda | \text{Data}) = \frac{\lambda^{\alpha^*-1}}{\Gamma(\alpha^*) \beta^{*\alpha^*}} \exp \left\{ -\lambda/\beta^* \right\}.$$

The posterior distributions of $\delta | \lambda$ and λ are respectively the normal and gamma distributions. The updated values of hyper-parameters are:

$$\xi^* = B/A, \quad \sigma^* = A^{-1/2}, \quad (27.16)$$

$$\alpha^* = \alpha + k/2 \quad \beta^* = (1/\beta + 1/D)^{-1} \quad (27.17)$$

with

$$A = \sum_{i=1}^k \Delta x_i + \frac{1}{\sigma^2} \quad B = \sum_{i=1}^k \Delta t_i + \frac{\xi}{\sigma^2}$$

$$C = \sum_{i=1}^k \frac{(\Delta t_i)^2}{\Delta x_i} + \frac{\xi^2}{\sigma^2} \quad D = \frac{1}{2} \left(C - \frac{B^2}{A} \right).$$

The Bayesian procedure can update the joint probability law of δ and λ using new observations. The hyper-parameters are modified sequentially. After i -th update, they will be written in the following as $\alpha^{(i)}, \beta^{(i)}, \xi^{(i)}$ and $\sigma^{(i)}$ and the corresponding prior distribution will be $f^{(i)}(\delta, \lambda) = f_1^{(i)}(\delta | \lambda) f_2^{(i)}(\lambda)$. The values of hyper-parameters related to initial prior are $\alpha^{(0)}, \beta^{(0)}, \xi^{(0)}$ and $\sigma^{(0)}$.

27.4.2 Maintenance and Inspections Decision Rule

Let's consider a predictive maintenance policy with aperiodic inspection planning. Such a policy allows adapting the time between inspections to the current degradation level of the maintained system. When the system is inspected, the next inspection time can be determined. As a predictive policy, it is based on the system's RUL described in Sect. 27.2.2. Let $\{T_n\}_{n \in \mathbb{N}}$ denote the sequence of inspection times, with $T_0 = 0$. Consider:

$$T_{n+1} = T_n + \tau_p(X_{T_n}),$$

where $\tau_p(X_{T_n})$ is defined from Eq. (27.9) as the p -quantile of the remaining useful life distribution given in Eq. (27.14), i.e.,

$$\tau_p(X_{T_n}) = \{\Delta t : F_{R_k}(\Delta t | X_{T_n} = x_n) = p\}. \quad (27.18)$$

Figures 27.3 and 27.4 show respectively the pdf and cdf of the system's RUL at inspection times between two replacements. The simulated sample path has been generated with the decision variable $p = 0.045$. It ends when the degradation level is greater than $M = 6.44$. As explained previously, the times between successive inspections are defined, and the degradation model is with unknown parameters with the same model as in Sect. 27.4.1. The degradation parameters' distributions are such that $\mathbb{E}[\lambda] = 1$, $\mathbb{V}[\lambda] = 0.1$, $\mathbb{E}[\delta] = 1$ and $\mathbb{V}[\delta] = 0.07$.

According to Sect. 27.2.2, T_n^- for $n \in \mathbb{N}$ refers to the inspection time just before possible maintenance action. The decision rule at each inspection time is as follows:

- If $X_{T_n^-} \geq L$, the system has failed and is correctively replaced.
- If $M \leq X_{T_n^-} < L$, the system has not yet failed but is considered as "close to failure". As a consequence, preventive action is taking place immediately.
- If $X_{T_n^-} < M$, the system is still properly functioning, and there is no need for replacement. The system is left as it is.

As preventive and corrective replacements are supposed to be perfect, the system degradation level after maintenance is 0. In other words, it is "as good as new." More precisely:

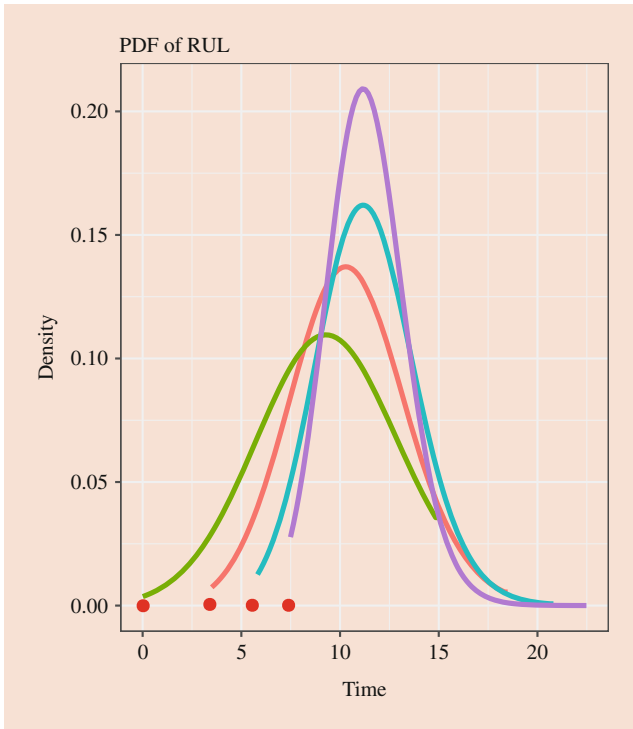


Fig. 27.3 The pdf of the system’s RUL at successive inspection times for a cycle path with $(T_1, X_{T_1}) = (3.5, 2.47)$, $(T_2, X_{T_2}) = (5.8, 3.9)$ and $(T_3, X_{T_3}) = (7.5, 5.54)$

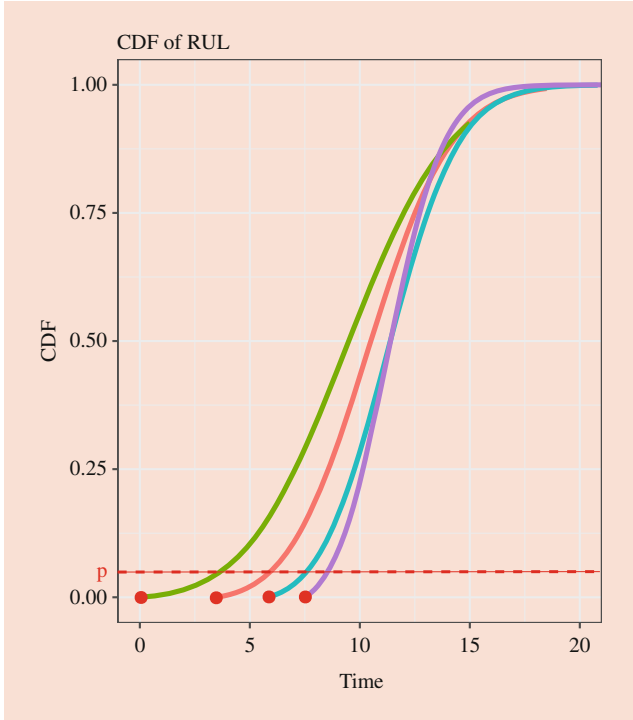


Fig. 27.4 The cdf of the system’s RUL at successive inspection times for a cycle (same path as in Fig. 27.3)

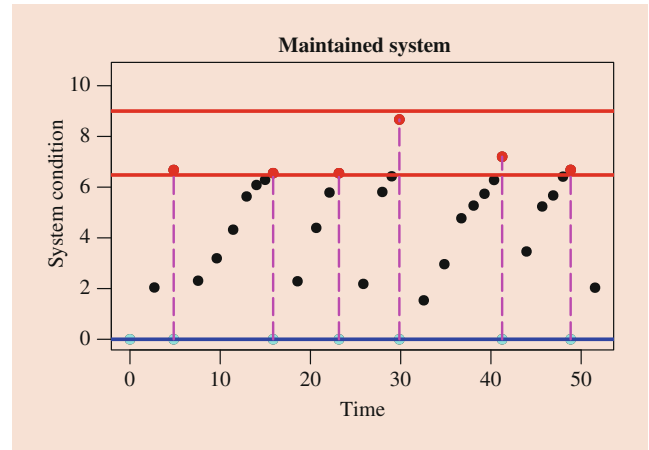


Fig. 27.5 The degradation path for a maintained system with unknown degradation parameters such that $\mathbb{E}[\delta] = 1$, $\mathbb{V}[\delta] = 0.5$, $\mathbb{E}[\lambda] = 1$ and $\mathbb{V}[\lambda] = 0.5$. Decision variables are $p = 0.09$, and $M = 6.48$

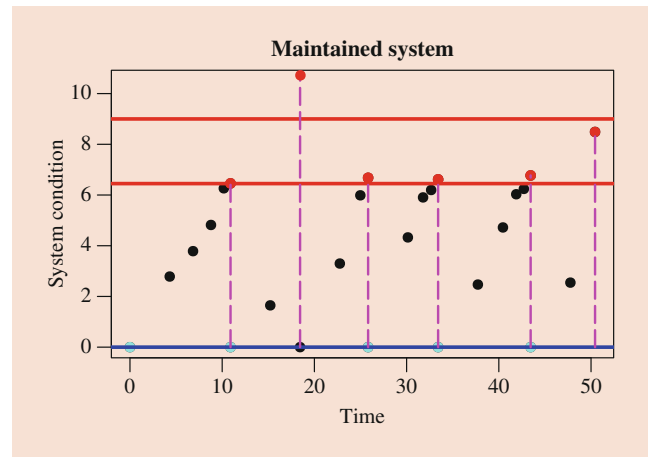


Fig. 27.6 The degradation path for a maintained system with known degradation parameters, $\delta = 1$ and $\lambda = 1$. Decision variables are $p = 0.035$ and $M = 6.46$

$$\begin{aligned} \text{If } X_{T_n} \geq L & \quad \text{then } X_{T_n} = 0 \\ \text{If } M \leq X_{T_n} < L & \quad \text{then } X_{T_n} = 0 \\ \text{If } 0 \leq X_{T_n} < M & \quad \text{then } X_{T_n} = X_{T_n} \end{aligned}$$

For more illustration, Figs. 27.5 and 27.6 represent the evolution of the degradation level of a component that is maintained according to the previous maintenance decision rule. For Fig. 27.5, the degradation parameters are unknown and described as in Sect. 27.4.1. Figure 27.6 is with known degradation parameters. As expected, lifetime modeling hence the inspection planning is more accurate when parameters are known. It can be seen that the mean number of inspections in a renewal cycle is higher in case of uncertainty on degradation parameters.

Finally, the considered decision rule has two decision variables: p and M . The decision-maker must determine these two variables to optimize a given assessment criterion. Classically, the economic criterion is considered, as defined in the next section. Note that in Figs. 27.5 and 27.6, optimal decision variables are considered. They have been optimized by minimization of the maintenance costs with the same unit costs.

27.4.3 Maintenance Cost

The inspections are planned sequentially, and each inspection incurs a unit cost denoted by C_i . Preventive and corrective actions are performed with respective costs C_p and C_c such that $C_c > C_p$. Failures are not self-announcing; hence, they can only be diagnosed and isolated at inspection times. Consequently, an additional cost at a rate of C_d is incurred from the failure time until the next replacement time. It is hereafter referred to as the unavailability cost rate. Let $C(t)$ define the cumulative maintenance cost up to time t . Considering that $N_i(t)$, $N_p(t)$ and $N_c(t)$ are, respectively, the numbers of inspections, preventive replacements, and corrective replacements in $[0, t]$, it comes:

$$C(t) = C_i N_i(t) + C_p N_p(t) + C_c N_c(t) + C_d d(t), \quad (27.19)$$

where $d(t)$ is the total time passed in a failed state in $[0, t]$.

As in the usual framework for dynamic maintenance, a long-run cost rate is considered to propose a tentative planning or decision rule settings for inspections and maintenance actions. It is intended to be applied on a rolling horizon basis and modified dynamically based on information obtained in real-time. The baseline objective cost function is then the long-term expected maintenance cost per time unit, which is defined by:

$$\begin{aligned} EC_\infty &= \lim_{t \rightarrow \infty} \frac{E[C(t)]}{t} \\ &= \lim_{t \rightarrow \infty} \frac{1}{t} E \left[\sum_{n \in \mathbb{N}} \mathbb{I}_{\{T_n < t\}} (1 + \mathbb{I}_{\{M \leq X_{T_n} < L\}} \right. \\ &\quad \left. + \mathbb{I}_{\{X_{T_n} \geq L\}}) + \int_0^t \mathbb{I}_{\{X_s \geq L\}} ds \right]. \end{aligned}$$

Note that mathematical derivation of such a criterion may take advantage of renewal theory. Notably, in the presence of perfect repair or replacement, the stochastic process describing the maintained system is regenerative. Renewal property allows rewriting the long-run average cost per time unit as the ratio of the expected cost on the first renewal cycle over this cycle's expected length. In other words:

$$EC_\infty = \frac{E[C(S_1)]}{E(S_1)}. \quad (27.20)$$

where S_1 is the first regeneration time, i.e., the time of first preventive or corrective replacement.

The decision variables for the maintenance policy are p and M . Tentative values can be obtained by minimization of the maintenance cost EC_∞ . The optimal decision variables, p^* and M^* , are such that:

$$\begin{aligned} EC_\infty(p^*, M^*) &\leq EC_\infty(p, M), \\ \forall (p, M) &\in [0, 1] \times [0, L], \end{aligned}$$

For an aperiodic maintenance policy with a gradual deterioration process, analytical expressions of EC_∞ cannot be derived easily from Eq. (27.20) because of the complexity of scenarios that can arise. The next section will focus on using of Markov renewal properties to overcome this difficulty.

27.4.4 Decision Variables Based on Long-Term Assessment

In this section, the maintained system's evolution is characterized to derive an expression for the long-run expected maintenance cost. As the degradation parameters are unknown, the calculation has to be done based on the available information at the time of calculation. As explained in Sect. 27.4.1, the prior distribution of δ and λ contains all this information.

Semi-regenerative Process and Cost Rate

The process $\{X_t\}_{t \geq 0}$ is a regenerative process, with regeneration times being the dates of replacement. Let's consider the discrete-time stochastic process describing the system state at inspection times and after maintenance action, if any. It is hereafter denoted $\{Y_n\}_{n \in \mathbb{N}}$ with $Y_n = X_{T_n}$. The embedded process $\{Y_n, T_n\}_{n \in \mathbb{N}}$ is a Markov renewal process, and the process $\{X_t\}_{t \geq 0}$ is a semi-regenerative process with semi-regeneration times being the inspection times. Due to the maintenance policy with perfect replacements, the embedded Markov chain $\{Y_n\}_{n \in \mathbb{N}}$ has $\{0\}$ as the regeneration set. Such a Markov chain is called Harris recurrent (or just Harris chain) in [67]. A stationary measure π can be defined, and it is unique up to a multiplicative constant; see [67] (pages 200-201). Then it comes (see [35]):

$$EC_\infty = \lim_{t \rightarrow \infty} \frac{E[C(t)]}{t} = \frac{E_\pi[C(T_1)]}{E_\pi[T_1]},$$

where T_1 is the first inspection time and \mathbb{E}_π refers to the expectation with respect to the stationary probability distribution π .

Note that the previous equality holds since the process $C(t)$ is such that:

- $C(t)$ is positive;
- $C(0) = 0$;
- $C(t)$ is given by $C(t) = \Psi_t(X_u, 0 \leq u \leq t)$ and is such that for $0 \leq s \leq t$, $C(t) - C(s) = \Psi_{t-s}(X_u, s \leq u \leq t)$ which is true for a cumulative cost.

With the classical renewal property, the expression of the maintenance cost EC_∞ can be obtained from the analysis of the system behavior on a regenerative cycle, i.e., between two replacements. The semi-regenerative property allows deriving EC_∞ from the analysis on a semi-regenerative cycle, i.e., between two successive inspections, which is simpler. As a prerequisite, this approach requires the knowledge of the stationary probability distribution π . It is derived from the transition probability density function of $\{Y_n\}_{n \in \mathbb{N}}$ which is developed in next paragraphs.

Degradation at Inspection Times

For all $n \in \mathbb{N}$, Y_n describes the system degradation level at n -th inspection time and after maintenance action if necessary. Thus, the process $\{Y_n\}_{n \in \mathbb{N}}$ takes values in $[0, M)$. In the following, $Y_n^- = X_{T_n^-}$ will denote the system degradation level at inspection time T_n^- , i.e., just before maintenance action.

First, consider the case that the two degradation parameters, δ and λ , are given. From Markovian properties and given the current state x with $x < M$, the probability of transition from state $Y_n = x$ at time T_n to state $Y_{n+1} = y$ at time T_{n+1} is given according to the maintenance decision rule. Two scenarios can arise if $Y_{n+1}^- \geq M$ the degradation level overpasses the maintenance threshold, and the system or component is replaced, i.e., $Y_{n+1} = 0$; if $Y_{n+1}^- < M$, only the next inspection is scheduled and $Y_{n+1} = Y_{n+1}^-$. It comes:

$$\begin{aligned} \Pr(Y_{n+1} = 0 | Y_n = x, \delta, \lambda) &= \bar{F}_{\tau_p(x)/\delta, \lambda \tau_p^2(x)}(M - x) \\ &\quad \text{(case } y = 0) \\ \Pr(Y_{n+1} \leq y | Y_n = x, \delta, \lambda) &= F_{\tau_p(x)/\delta, \lambda \tau_p^2(x)}(y - x) \\ &\quad \text{(case } x \leq y < M) \end{aligned}$$

with

$$\begin{aligned} &F_{\tau_p(x)/\delta, \lambda \tau_p^2(x)}(y - x) \\ &= \Pr(X_{t+\tau_p(x)} - X_t \leq y - x | X_t = x) \\ &= \int_0^{y-x} f_{\tau_p(x)/\delta, \lambda \tau_p^2(x)}(u) du \end{aligned}$$

and

$$\bar{F}_{\tau_p(x)/\delta, \lambda \tau_p^2(x)}(M - x) = 1 - F_{\tau_p(x)/\delta, \lambda \tau_p^2(x)}(M - x).$$

Consequently, the transition probability distribution function of $\{Y_n\}_{n \in \mathbb{N}}$ can be written as:

$$\begin{aligned} \Pr(dy|x, \delta, \lambda) &= \bar{F}_{\tau_p(x)/\delta, \lambda \tau_p^2(x)}(M - x) \delta_0(dy) \\ &\quad + f_{\tau_p(x)/\delta, \lambda \tau_p^2(x)}(y - x) \mathbb{I}_{\{x \leq y < M\}} dy, \end{aligned} \quad (27.21)$$

where δ_0 denotes the Dirac delta function.

As explained previously, the first part of the right-hand side in Eq. (27.21) is associated with replacements when the degradation level is higher than M at inspection time just before maintenance. The second part describes a classical degradation increment from x to y for $x \leq y$ and when $y < M$.

In the case of unknown degradation parameters, the transition probability density function of $\{Y_n\}_{n \in \mathbb{N}}$ is obtained from the transition pdf knowing the degradation parameters by:

$$\begin{aligned} \Pr(dy|x) &= \int_0^\infty \int_{-\infty}^{+\infty} \Pr(dy|x, \delta, \lambda) \\ &\quad f_1(\delta|\lambda) f_2(\lambda) d\delta d\lambda. \end{aligned} \quad (27.22)$$

Replacing $\Pr(dy|x, \delta, \lambda)$ by its expression from Eq.(27.21) leads to:

$$\begin{aligned} \Pr(dy|x) &= \bar{G}_{\tau_p(x)}(M - x) \delta_0(dy) \\ &\quad + g_{\tau_p(x)}(y - x) \mathbb{I}_{\{x \leq y < M\}} dy, \end{aligned} \quad (27.23)$$

where

$$\begin{aligned} g_{\tau_p(x)}(z) &= \sqrt{\frac{\beta}{2\pi}} \frac{\Gamma(\alpha + 1/2) \tau_p(x)}{\Gamma(\alpha)} z^{-3/2} \\ &\quad \times (\sigma^2 z + 1)^{-1/2} \left(1 + \frac{\beta(\xi z - \tau_p(x))^2}{2z(\sigma^2 z + 1)} \right)^{-(\alpha+1/2)}, \end{aligned}$$

and

$$\bar{G}_{\tau_p(x)}(z) = \int_z^\infty g_{\tau_p(x)}(u) du.$$

Stationary Law

The unique stationary probability distribution π of the Markov chain $\{Y_n\}_{n \in \mathbb{N}}$ has the same structure as $\Pr(dy|x)$ with a mass and a density part, i.e.:

$$\pi(dx) = \tilde{a}_1 \delta_0(dx) + \tilde{a}_2 b(x) dx. \quad (27.24)$$

Besides, the stationary law is a solution of the stationary equation:

$$\pi(dy) = \int_{[0,M]} \Pr(dy|x)\pi(dx). \quad (27.25)$$

Let replace, respectively, $\Pr(dy|x)$ and $\pi(\cdot)$ in (27.25) by their expressions from Eqs. (27.23) and (27.24). The density part of the resulting equation can be written for $y < M$:

$$\tilde{a}_2 b(y) = \tilde{a}_1 g_{\tau_p(0)}(y) + \tilde{a}_2 \int_0^y g_{\tau_p(x)}(y-x)b(x)dx. \quad (27.26)$$

and as $\int_{[0,M]} \pi(dx) = 1$, it can be shown that $\tilde{a}_1 = 1 - \tilde{a}_2$. If $B(y)$ is defined for any $y \in [0, M]$ by:

$$B(y) = \frac{1 - \tilde{a}_1}{\tilde{a}_1} b(y), \quad (27.27)$$

Equation (27.26) can be rewritten:

$$B(y) = g_{\tau_p(0)}(y) + \int_0^y B(x)g_{\tau_p(x)}(y-x)dx. \quad (27.28)$$

The resulting equation is the Volterra equation of the second kind. Then, $B(\cdot)$ can be obtained numerically through classical numerical schemes; see, for example, [68]. Finally, from $\int_{[0,M]} \pi(dx) = 1$ and the expression of $\pi(dx)$ in Eq. (27.24), it comes:

$$\frac{1}{\tilde{a}_1} = 1 + \int_{[0,M]} B(x)dx,$$

hence

$$\tilde{a}_1 = \frac{1}{1 + \int_0^M B(x)dx}. \quad (27.29)$$

As an illustration, Fig. 27.7 shows the density functions of the stationary probability distribution π obtained for four different configurations of degradation parameters. The failure threshold is $L = 9$ and the decision variables are set to $p = 0.036$ and $M = 6.45$ for all cases. Configurations considered in Fig. 27.7 are of two types:

- (i) The case of known degradation parameters with $\lambda = 1$ and $\delta = 1 (= 1/\mu)$, and
- (ii) For unknown parameters, three different prior distributions are considered, as given in Table 27.1.

The probability density functions of priors given in Table 27.1 are depicted, respectively, in Figs. 27.8, 27.9 and 27.10. The expected values of δ and λ are identical for the three cases and equal to the values considered for known parameters.

Figure 27.7 shows, for fixed decision variables, that the probability of observing a low degradation level (less than 3) at inspection times decreases when the uncertainty of degradation parameters decreases.

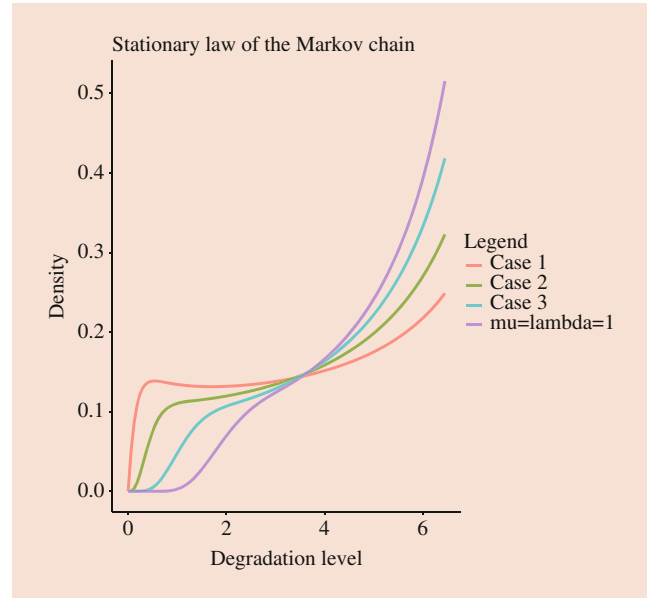


Fig. 27.7 Stationary probability distributions for known and unknown degradation parameters with $L = 9$, $p = 0.036$, and $M = 6.45$

Table 27.1 Configurations for prior distribution considered on Fig. 27.7

	Case 1	Case 2	Case 3
$E[\lambda]$	1	1	1
$V[\lambda]$	0.5	0.2	0.1
$E[\delta]$	1	1	1
$V[\delta]$	0.5	0.2	0.07

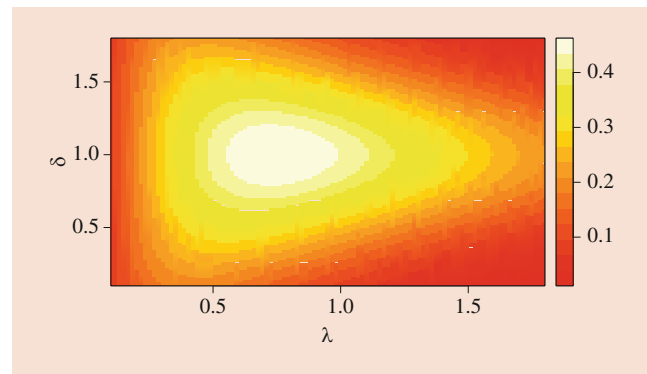


Fig. 27.8 Prior distributions of degradation parameters: Case 1

Long-Run Cost Rate

Finally, the long-term expected maintenance cost of the system per unit of time can be assessed by expectation with respect to π as:

$$EC_\infty = \lim_{t \rightarrow \infty} \frac{E[C(t)]}{t} = \frac{E_\pi[C(T_1)]}{E_\pi[T_1]}, \quad (27.30)$$

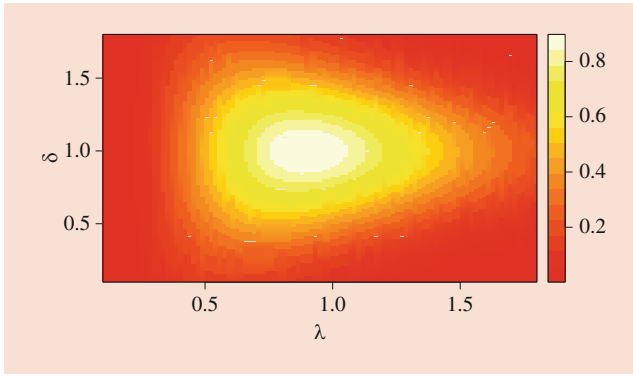


Fig. 27.9 Prior distributions of degradation parameters: Case 2

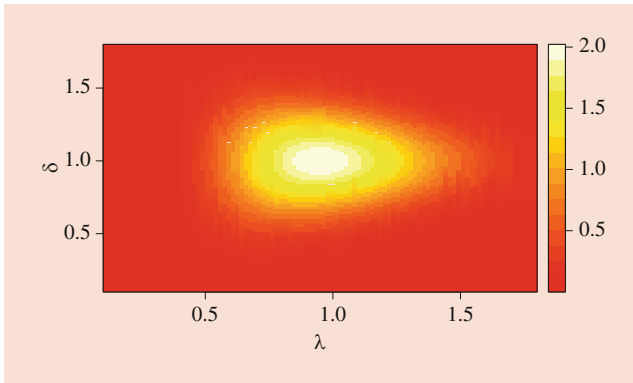


Fig. 27.10 Prior distributions of degradation parameters: Case 3

where $C(t)$ is defined in Eq. (27.19). Obviously, $E_{\pi}[N_i(T_1)] = 1$. The other expectations are given by:

$$E_{\pi}(N_p(T_1)) = P_{\pi}(M \leq X_{T_1^-} < L) = \int_0^M [\bar{G}_{\tau_p(x)}(M - x) - \bar{G}_{\tau_p(x)}(L - x)] \pi(dx),$$

$$E_{\pi}(N_c(T_1)) = P_{\pi}(X_{T_1^-} \geq L) = \int_0^M \bar{G}_{\tau_p(x)}(L - x) \pi(dx),$$

$$E_{\pi}(d(T_1)) = \int_0^M \left[\int_0^{\tau_p(x)} \bar{G}_s(L - x) ds \right] \pi(dx),$$

and finally:

$$E_{\pi}(T_1) = \int_0^M \tau_p(x) \pi(dx).$$

These integrals can be calculated using standard numerical integration techniques like the trapezoidal rule or Simpson's rule.

27.4.5 Dynamic Policy

The long-run expected maintenance cost for a given level of information about degradation parameters is developed in the previous section. It is then possible to optimize the decision variables p and M with respect to that cost. This section describes the dynamic procedure to adjust the maintenance decision rule parameters according to updated information. In the dynamic framework, decision parameters are derived from maintenance costs assessed on a long-term horizon. They are considered on a short-term rolling horizon. New information obtained on that short-term horizon is included and helps to update the expression of long-term maintenance cost hence the values of decision parameters for the next planning horizon. The planning horizon hereafter is the length of a renewal cycle, i.e., the time to the next perfect repair or replacement.

Algorithm

Let $\Theta^{(i)} = (\alpha^{(i)}, \beta^{(i)}, \xi^{(i)}, \sigma^{(i)})$ denote the values of hyper-parameters after i -th update, starting from the initial prior for $i = 0$. The function $f^{(i)}(\delta, \lambda)$ is the joint prior distribution of (δ, λ) obtained with values $\Theta^{(i)}$ of hyperparameters. The algorithm for maintenance decision-making is as follows:

Step 0: Initialization. Set $i = 0, k = 0, X_k = 0$, and $T_k = 0$. Choose the initial values of the hyper-parameters $\Theta^{(0)}$.

Step 1: Optimize the value of decision variables for the renewal cycle i by minimizing the cost rate given in Eq. (27.30). The long-run cost rate optimization requires assessments of the stationary law $\pi^{(i)}$, which is obtained from Eq. (27.24) with hyper-parameters $\Theta^{(i)}$.

Step 2: Clean up the memory of degradation increments. Apply the aperiodic maintenance policy for the current cycle:

- **2.1:** Evaluate $\Delta T_k = \tau_p(X_{T_k})$ according to the RUL distribution with $\Theta^{(i)}$, see Eq. (27.14).
- **2.2:** Inspect the system at time $T_{k+1} = T_k + \Delta T_k$.
- **2.3:** If $X_{k+1} < M$, then memorize the degradation increment $\Delta X_{k+1} = X_{T_{k+1}} - X_{T_k}$ and the corresponding time interval ΔT_k . Set $k = k + 1$, and go back to sub-step 2.1.
 Else if $X_{k+1} \geq M$ then let $X_{k+1} = 0$ (i.e., maintain the system) and set $k = k + 1$.

Step 3: Update the degradation model from observations during the last cycle, i.e., find the new values of hyper-parameters $\Theta^{(i+1)}$ from all the degradation increments which are in memory, see Eq. (27.16). Set $i = i + 1$ and go to step 1.

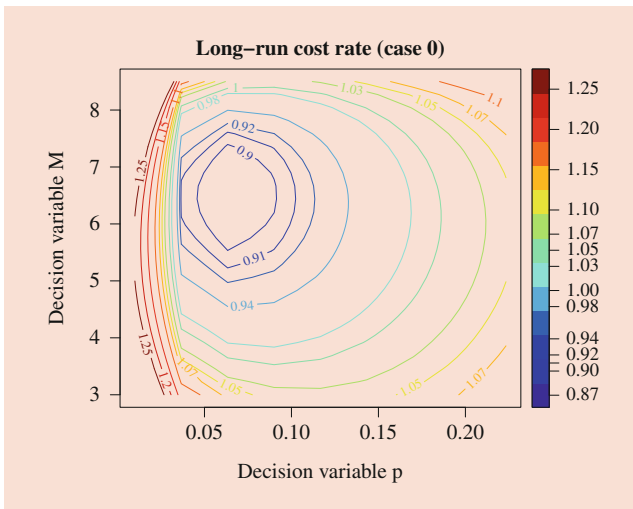


Fig. 27.11 Level curves of long-run cost rate with $\mathbb{E}[\delta] = 1.1$, $\mathbb{V}[\delta] = 1.1$, $\mathbb{E}[\lambda] = 1$ and $\mathbb{V}[\lambda] = 1$

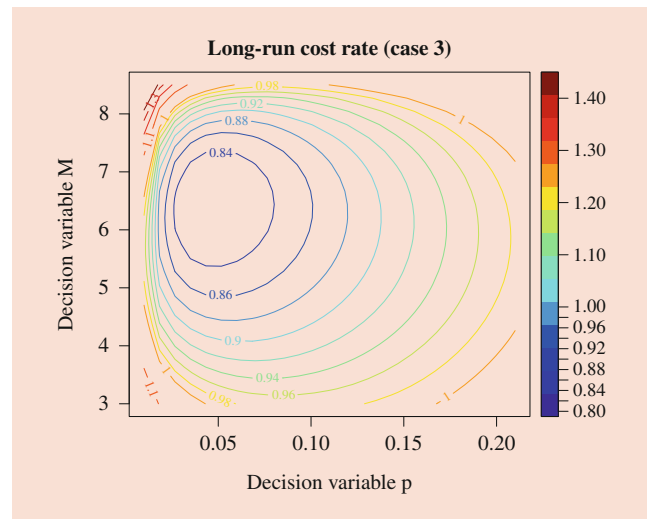


Fig. 27.13 Level curves of long-run cost rate for case 3 from Table 27.1

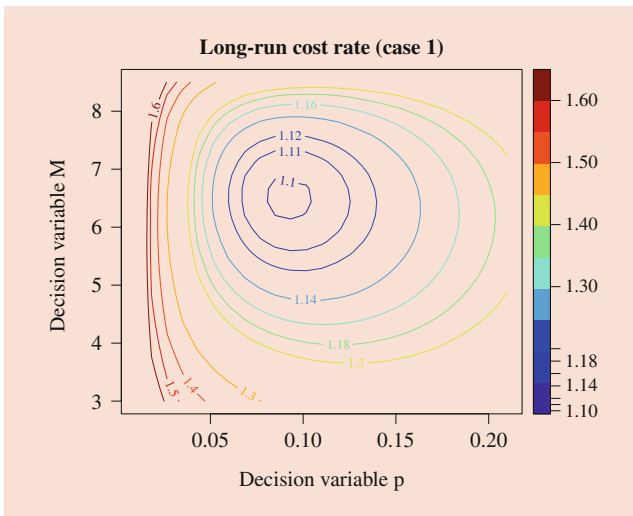


Fig. 27.12 Level curves of long-run cost rate for case 1 from Table 27.1

As stated above, each step of the dynamic process requires the minimization of the long-run cost rate for hyper-parameters current values. No analytical expression is available for optimal values. The long-run cost rate must have a unique minimum to ensure that such optimal values can be determined without ambiguity. Numerous numerical experiments have been carried out for a wide range of hyper-parameters. They all show convex shapes for the cost as a function of decision variables. For illustration purposes, three examples of cost rate functions are given in Figs. 27.11, 27.12, and 27.13 for different sets of hyper-parameters with decreasing variance for degradation

parameters δ and λ as happening during the dynamic process. The variance of δ is successively 1.1, 0.5 and 0.07 and these values for λ are, respectively, 1, 0.5, and 0.1.

Due to observed convexity, classical nonlinear optimization algorithms can be used for numerical optimization. The downhill simplex method proposed in [69], also known as Nelder-Mead method, has been considered for optimization in this chapter. This method requires a significant number of successive numerical evaluations of the cost function during the optimization process. As a consequence, it is not very efficient in terms of computing speed. However, it does not require derivatives evaluations and can be robust [68].

Behavior of the Dynamic Policy

A numerical study and analysis of such a dynamic maintenance policy can be found in [47], as well as its comparison with other policies. It shows the relevance of dynamic strategies that are based on long-term cost rates with updates from monitoring information. Figures 27.14 and 27.15 give examples of inspection times and related degradation levels for two levels of information based on cases 1 and 3 considered previously. It can be seen that as the uncertainty about degradation parameters decreases, from case 1 to case 3, the number of inspections in a renewal cycle decreases.

Figure 27.16 is an example of the mean evolution of two decision variables, p and M , after successive updates. It is based on the simulation of 50 paths from time $t = 0$ to 100. The main modifications of the values take place in the first quarter of the sequence. The information collected online leads to significant updates and substantial effects on the evolution of parameters p and M .

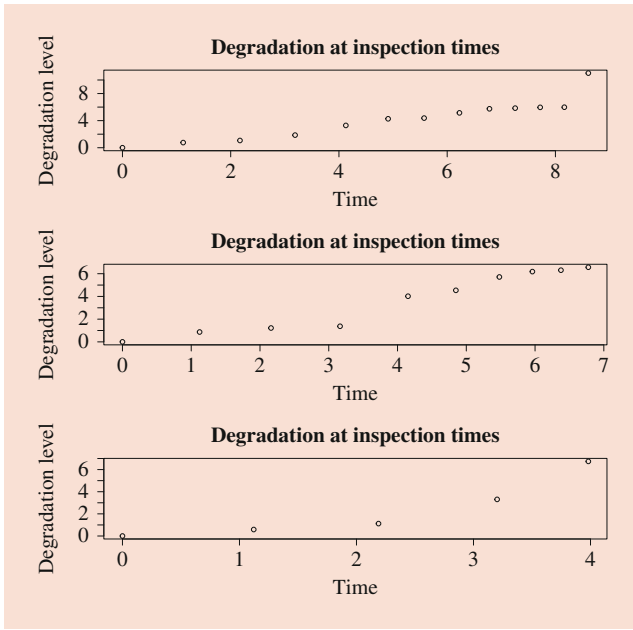


Fig. 27.14 Inspection times and degradation levels on one cycle: Case 1

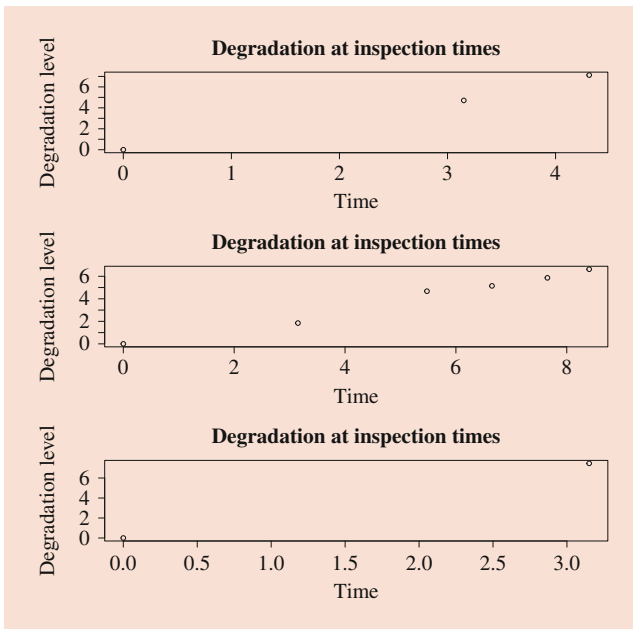


Fig. 27.15 Inspection times and degradation levels on one cycle: Case 3

27.5 Conclusion

This chapter is related to dynamic condition-based and predictive maintenance modeling. It focuses on the case of continuous degradation and aims to describe a dynamic decision-making framework’s characteristics. Classical stochastic processes for continuous degradation modeling are firstly recalled with their main indicators for failure

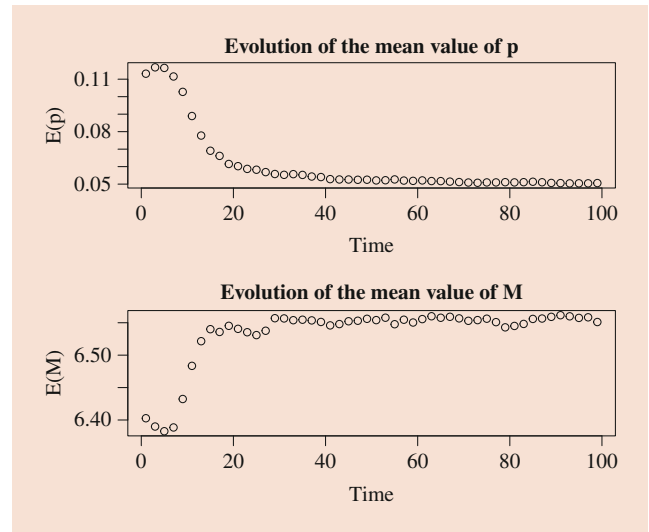


Fig. 27.16 Example of dynamic evolution of decision variables

prediction. The most common structure of the maintenance decision rule is also given. The context resulting in a need for a dynamic approach is then described in two main directions. The first one is related to maintenance grouping in multicomponent systems, with opportunistic maintenance and (economic) dependencies. The second one deals with uncertainty in degradation description or behavior. Information collected in real time is used to update the planning or improve the degradation model and the maintenance decision rule. The general framework of dynamic decision-making is described with successive cycles on a finite time-rolling horizon. On each cycle, the starting point for maintenance and inspection planning is based on long-term system behavior. A long-term cost rate is optimized to determine tentative planning for each component or the decision rule variables. In the case of maintenance grouping, this planning may be modified with a short-term criterion. The availability of new information leads to an update of the maintenance model and the beginning of a new cycle. Finally, a specific case related to gradual degradation modeling with unknown parameters is described precisely as an example of dynamic condition-based predictive maintenance.

References

1. Zhang, Q., Basseville, M., Benveniste, A.: Early Warning of Slight Changes in Systems and Plants with Application to ConditionBased Maintenance RR-1750, (1992)
2. McCall, J.: Maintenance policies for stochastically failing equipment : a survey. *Manag. Sci.* **11**(5), 493–524 (1965)
3. Pierskalla, W., Voelker, J.: A survey of maintenance models : the control and surveillance of deteriorating systems. *Naval Res. Logist. Q.* **23**, 353–388 (1979)

4. Sherif, Y, Smith, M: Optimal maintenance models for systems subject to failure : a review. *Naval Res. Logist. Q.* **28**, 47–74 (1981)
5. Thomas, L: A survey of maintenance and replacement models for maintainability and reliability of multi-item systems. *Reliab. Eng.* **16**, 297–309 (1986)
6. Valdez-Flores, C., Feldman, R.: A survey of preventive maintenance models for stochastically deteriorating single-unit systems. *Naval Res. Logist.* **36**, 419–446 (1989)
7. Barlow, R., Proschan, F.: *Mathematical Theory of Reliability, Classics in Applied Mathematics*, vol. 17. SIAM, Society for Industrial and Applied Mathematics, Philadelphia (1996)
8. van der Duyn Schouten, F.: Maintenance policies for multicomponent systems : an overview. In: Özekici, S. (ed.) *Reliability and Maintenance of Complex Systems*, NATO ASI Series—Series F: Computer and Systems Sciences, vol. 154, pp. 118–136. Springer, Berlin (1996)
9. Dekker, R., van der Duyn Schouten, F., Wildeman, R.: A review of multi-component maintenance models with economic dependence. *Math. Methods Oper. Res.* textbf45, 411–435 (1997)
10. Wang, H.: A survey of maintenance policies of deteriorating systems. *Eur. J. Oper. Res.* **139**, 469–489 (2002). [https://doi.org/10.1016/S0377-2217\(01\)00197-7](https://doi.org/10.1016/S0377-2217(01)00197-7)
11. Shafiee, M., Chukova, S.: Maintenance models in warranty: a literature review. *Eur. J. Oper. Res.* **229**(3), 561–572 (2013). <https://doi.org/10.1016/j.ejor.2013.01.017>
12. Zhao, X., Al-Khalifa, K., Magid Hamouda, A., Nakagawa, T.: Age replacement models: a summary with new perspectives and methods. *Reliab. Eng. Syst. Saf.* **161**, 95–105 (2017)
13. Alaswad, S., Xiang, Y.: A review on condition-based maintenance optimization models for stochastically deteriorating system. *Reliab. Eng. Syst. Saf.* **157**(Supplement C), 54–63 (2017). <https://doi.org/10.1016/j.res.2016.08.009>
14. Grall, A., Bérenguer, C., Dieulle, L.: A condition-based maintenance policy for a stochastically deteriorating system. *Reliab. Eng. Syst. Saf.* **76**(2), 167–180 (2002)
15. Tan, L., Cheng Z., Guo B., Gong, S.: Condition-based maintenance policy for gamma deteriorating systems. *J. Syst. Eng. Electron.* **21**(1), 57–61 (2010)
16. Mercier, S., Castro, I.T.: On the modelling of imperfect repairs for a continuously monitored gamma wear process through age reduction. *J. Appl. Probab.* **50**(4), 1057–1076 (2013). <https://doi.org/10.1239/jap/1389370099>
17. van Noortwijk, J.: A survey of the application of gamma processes in maintenance. *Reliab. Eng. Syst. Saf.* **94**, 2–21 (2009). <https://doi.org/10.1016/j.res.2007.03.019>
18. Chen, N., Ye, Z.-S., Xiang, Y., Zhang, L.: Condition-based maintenance using the inverse Gaussian degradation model. *Eur. J. Oper. Res.* **243**(1), 190–199 (2015)
19. Li, R., Liu, X., Song, Y., Cai, Z., Li, J.: An optimal condition based maintenance strategy using inverse-Gaussian degradation process. In: 2017 Second International Conference on Reliability Systems Engineering (ICRSE), pp. 1–5 (2017). <https://doi.org/10.1109/ICRSE.2017.8030762>
20. Wang, C., Xu, J., Wang, H., Zhang, Z.: Condition-based predictive order model for a mechanical component following inverse gaussian degradation process. *Math. Probl. Eng.* **2018** (2018)
21. Barker, C., Newby, M.: Optimal non-periodic inspection for multivariate degradation model. *Reliab. Eng. Syst. Saf.* **94**(1), 33–43 (2009)
22. Guo, C., Wang, W., Guo, B., Si, X.: A maintenance optimization model for mission-oriented systems based on Wiener degradation. *Reliab. Eng. Syst. Saf.* **111**, 183–194 (2013)
23. Elwany, A.H., Gebraeel, N.Z., Maillart, L.M.: Structured replacement policies for components with complex degradation processes and dedicated sensors. *Oper. Res.* **59**(3), 684–695 (2011)
24. Liu, B., Do, P., Iung, B., Xie, M.: Stochastic filtering approach for condition-based maintenance considering sensor degradation. *IEEE Trans. Autom. Sci. Eng.* **17**(1):177–190 (2019)
25. Gertsbakh, I.: *Reliability Theory - With Applications to Preventive Maintenance*. Springer, Berlin (2000)
26. Rao, B. (ed.): *Handbook of Condition Monitoring*. Elsevier, Amsterdam (1996)
27. de Jonge, B., Teunter, R., Tinga, T.: The influence of practical factors on the benefits of condition-based maintenance over time-based maintenance. *Reliab. Eng. Syst. Saf.* **158**, 21–30 (2017)
28. Whitmore, G., Seshadri, V.: A heuristic derivation of the inverse Gaussian distribution. *Am. Stat.* **41**(4), 280–281 (1987)
29. Park, C., Padgett, W.: Accelerated degradation models for failure based on geometric Brownian motion and gamma processes. *Life-time Data Anal.* **11**(4), 511–527 (2005)
30. Wang, X., Xu, D.: An inverse Gaussian process model for degradation data. *Technometrics* **52**(2), 188–197 (2010)
31. Ye, Z.-S., Chen, N.: The inverse Gaussian process as a degradation model. *Technometrics* **56**(3), 302–311 (2014)
32. Dieulle, L., Bérenguer, C., Grall, A., Roussignol, M.: Sequential condition-based maintenance scheduling for a deteriorating system. *Eur. J. Oper. Res.* **150**(2), 451–461 (2003)
33. Zhao, X., Fouladirad, M., Bérenguer, C., Bordes, L.: Nonperiodic inspection/replacement policy for monotone deteriorating system with covariates. In: *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes Barcelona, Spain, June 30–July 3, 2009*, pp. 1617–1622. IFAC (2009)
34. Mercier, S., Meier-Hirmer, C., Roussignol, M.: Bivariate Gamma wear processes for track geometry modelling, with application to intervention scheduling. *Struct. Infrastruct. Eng.* **8**(4), 357–366 (2012). <https://doi.org/10.1080/15732479.2011.563090>
35. Grall, A., Dieulle, L., Bérenguer, C., Roussignol, M.: Continuous-time predictive-maintenance scheduling for a deteriorating system. *IEEE Trans. Reliab.* **51**(2), 141–150 (2002) <https://doi.org/10.1109/TR.2002.1011518>
36. Do, P., Voisin, A., Levrat, E., Iung, B.: A proactive condition-based maintenance strategy with both perfect and imperfect maintenance actions. *Reliab. Eng. Syst. Saf.* **133**, 22–32 (2015)
37. Khoury, E., Grall, A., Bérenguer, C.: A comparison of RUL-based and deterioration-based maintenance policies for gradually deteriorating systems. In: Ale, B., Papazoglou, I., Zio, E. (eds.) *Reliability, Risk and Safety: Back to the Future. Proceedings of the European Safety and Reliability Conference, ESREL 2010, Rhodes, Greece, 5–9 september 2010*, pp. 530–537. Taylor & Francis Group, London (2010)
38. Castanier, B., Bérenguer, C., Grall, A.: A sequential condition-based repair/replacement policy with non-periodic inspections. *Appl. Stoch. Models Bus. Ind.* **19**(4), 327–347 (2003). <https://doi.org/10.1002/asmb.493>
39. Chen, Y., Gong, W., Xu, D., Kang, R.: Imperfect maintenance policy considering positive and negative effects for deteriorating systems with variation of operating conditions. *IEEE Trans. Autom. Sci. Eng.* **15**(2), 872–878 (2017)
40. Olde Keizer, M.C.A., Flapper, S.D.P., Teunter, R.H.: Condition-based maintenance policies for systems with multiple dependent components: a review. *Eur. J. Oper. Res.* **261**(2), 405–420 (2017)
41. Do Van, P., Barros, A., Bérenguer, C., Bouvard, K., Brissaud, F.: Dynamic grouping maintenance with time limited opportunities. *Reliab. Eng. Syst. Saf.* **120**, 51–59 (2013) <https://doi.org/10.1016/j.res.2013.03.016>
42. Wildeman, R., Dekker, R., Smit, A.: A dynamic policy for grouping maintenance activities. *Eur. J. Oper. Res.* **99**, 530–551 (1997)
43. Vu, H.C., Do, P., Barros, A., Bérenguer, C.: Maintenance planning and dynamic grouping for multi-component systems with positive

- and negative economic dependencies. *IMA J. Manag. Math.* **26**(2), 145–170 (2015)
44. Bouvard, K., Artus, S., Bérenguer, C., Cocquempot, V.: Condition-based dynamic maintenance operations planning & grouping. *Appl. Commer. Heavy Veh. Reliab. Eng. Syst. Saf.* **96**(6), 601–610 (2011)
 45. Horenbeek, A.V., Pintelon, L.: A dynamic predictive maintenance policy for complex multi-component systems. *Reliab. Eng. Syst. Saf.* **120**, 39–50 (2013)
 46. Kallen, M.-J., van Noortwijk, J.M.: Optimal maintenance decisions under imperfect inspection. *Reliab. Eng. Syst. Saf.* **90**(2–3), 177–185 (2005)
 47. Mosayebi Omshi, E., Grall, A., Shemehsavar, S.: A dynamic auto-adaptive predictive maintenance policy for degradation with unknown parameters. *Eur. J. Oper. Res.* **282**(1), 81–92 (2020). <https://doi.org/10.1016/j.ejor.2019.08.050>
 48. Paroissin, C.: Online estimation methods for the gamma degradation process. *IEEE Trans. Reliab.* **66**(4), 1361–1367 (2017)
 49. Gebrael, N.Z., Lawley, M.A., Li, R., Ryan, J.K.: Residual-life distributions from component degradation signals: a Bayesian approach. *IIE Trans.* **37**(6), 543–557 (2005)
 50. Xua, W., Wangb, W.: RUL estimation using an adaptive inverse Gaussian model. *Chem. Eng.* **33**, 331–336 (2013)
 51. Bian, L., Gebrael, N.: Computing and updating the first-passage time distribution for randomly evolving degradation signals. *IIE Trans.* **44**(11), 974–987 (2012)
 52. Liu, B., Lin, J., Zhang, L., Kumar, U.: A dynamic prescriptive maintenance model considering system aging and degradation. *IEEE Access* **7**, 94931–94943 (2019)
 53. Flage, R., Coit, D.W., Luxhøj, J.T., Aven, T.: Safety constraints applied to an adaptive Bayesian condition-based maintenance optimization model. *Reliab. Eng. Syst. Saf.* **102**, 16–26 (2012)
 54. Mosayebi Omshi, E., Grall, A., Shemehsavar, S.: Bayesian update and aperiodic maintenance policy for deteriorating systems with unknown parameters. In: H. et al. (eds.), *Proceedings of the 28th European Safety and Reliability Conference, ESREL 2018, Trondheim, Norway, 17–21 June 2018. Safety and Reliability—Safe Societies in a Changing World*, pp. 687–692. Taylor & Francis Group, London (2018)
 55. Juang, M.-G., Anderson, G.: A Bayesian method on adaptive preventive maintenance problem. *Eur. J. Oper. Res.* **155**(2), 455–473 (2004)
 56. Sheu, S.-H., Yeh, R.H., Lin, Y.-B., Juang, M.-G.: A Bayesian approach to an adaptive preventive maintenance model. *Reliab. Eng. Syst. Saf.* **71**(1), 33–44 (2001)
 57. Fouladirad, M., Grall, A.: Condition-based maintenance for a system subject to a non-homogeneous wear process with a wear rate transition. *Reliab. Eng. Syst. Saf.* **96**(6), 611–618 (2011)
 58. Fouladirad, M., Grall, A., Dieulle, L.: On the use of on-line detection for maintenance of gradually deteriorating systems. *Reliab. Eng. Syst. Saf.* **93**(12), 1814–1820 (2008)
 59. Fouladirad, M., Grall, A.: Monitoring and condition-based maintenance with abrupt change in a system's deterioration rate. *Int. J. Syst. Sci.* **46**(12), 2183–2194 (2015)
 60. Fouladirad, M., Grall, A.: On-line change detection and condition-based maintenance for systems with unknown deterioration parameters. *IMA J. Manag. Math.* **25**(2), 139–158 (2014)
 61. Zhang, M., Gaudoin, O., Xie, M.: Degradation-based maintenance decision using stochastic filtering for systems under imperfect maintenance. *Eur. J. Oper. Res.* **245**(2), 531–541 (2015)
 62. Zhao, X., Liang, Z., Parlikad, A.K., Xie, M.: Dynamic imperfect condition-based maintenance for systems subject to nonlinear degradation paths (2018)
 63. Shen, J., Cui, L., Ma, Y.: Availability and optimal maintenance policy for systems degrading in dynamic environments. *Eur. J. Oper. Res.* **276**(1), 133–143 (2019)
 64. Huynh, K.T.: A hybrid condition-based maintenance model for deteriorating systems subject to nonmemoryless imperfect repairs and perfect replacements. *IEEE Trans. Reliab.* **69**(2), 781–815 (2019)
 65. Huynh, K.T.: Modeling past-dependent partial repairs for condition-based maintenance of continuously deteriorating systems. *Eur. J. Oper. Res.* **280**(1), 152–163 (2020)
 66. Peng, C.-Y.: Inverse Gaussian processes with random effects and explanatory variables for degradation data. *Technometrics* **57**(1), 100–111 (2015)
 67. Asmussen, S.: *Applied Probability and Queues, Applications of Mathematics*, 2nd edn. Springer, New York (2003)
 68. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: *Numerical Recipes: The Art of Scientific Computing*, 3rd edn. Cambridge University Press, Cambridge (2007)
 69. Nelder, J., Mead, R.: A simplex method for function minimization. *Comput. J.* **7**, 308–313 (1965)



Antoine Grall is a full professor of mathematics, reliability, and maintenance engineering at the University of Technology of Troyes, France. He received his Ph.D. in applied mathematics from the University of Technology of Compiègne, France. His research interests include stochastic degradation modelling, lifetime prognosis, condition-based and predictive maintenance policies especially with dynamic strategies using online monitoring information.

Elham Mosayebi Omshi is an assistant professor of statistics at University of Technology of Troyes, France. She received her Ph.D. in Statistics from the University of Tehran, Iran. Her current research interests include stochastic degradation modeling, maintenance and optimization, and reliability. She is interested in employing Bayesian statistical methods to improve modeling in the mentioned topics.



Stochastic Redundant Replacement Maintenance Models

28

Toshio Nakagawa , Satoshi Mizutani , and Xufeng Zhao

Contents

28.1 Introduction	543
28.2 Single Policy	545
28.2.1 Age Replacement.....	545
28.2.2 Periodic Replacement.....	546
28.3 Two Policies	547
28.3.1 Age Replacement.....	547
28.3.2 Periodic Replacement.....	552
28.4 Three Policies	560
28.4.1 Replacement First and Last.....	560
28.4.2 Replacement Middle.....	561
28.4.3 Modified Replacement First and Last.....	561
28.5 Redundant Replacement Policies	562
28.5.1 Age Replacement.....	563
28.5.2 Periodic Replacement.....	564
Appendix 1.....	565
Appendix 2.....	565
Appendix 3.....	566
Appendix 4.....	566
Appendix 5.....	567
Appendix 6.....	567
Appendix 7.....	568
Appendix 8.....	569
Appendix 9.....	569
Appendix 10.....	569
Appendix 11.....	570
References	570

Abstract

Many serious accidents have happened as systems have become large scale and complex, and moreover, advanced nations have almost finished infrastructures and rushed into a maintenance period. Maintenance would be more important than production and construction for environment consideration and the protection of natural resources. A variety of maintenance policies have been established to prevent failures for objective systems in reliability theory. It has been well-known that high system reliability can be achieved through redundancy. Alternatively, several maintenance policies are planned simultaneously, such as bivariate, trivariate, and multivariate policies with multiple maintenance plans. This chapter takes up age and periodic replacements that are the most standard maintenance policy, shows their optimal policies, and proposes redundant replacement policies with time T and n kinds of replacements. The results obtained in this chapter would be applied to maintenances for redundant systems, imperfect repair, and several failure nodes.

Keywords

Reliability · Maintenance · Age replacement · Periodic replacement · Replacement first · Replacement last · Replacement overtime · Redundant maintenance

28.1 Introduction

Performance degradation and replacement strategy for manufacturing systems are commonly encountered in practice. Replacements done after failure and before failure are called corrective replacement and preventive replacement, respectively [1]. Age and periodic replacement policies have been modelled extensively in maintenance engineering. Age replacement is a time-based policy in which an operating unit is replaced by a new one or is perfectly repaired and becomes

T. Nakagawa (✉) · S. Mizutani
 Department of Business Administration, Aichi Institute of Technology,
 Aichi, Japan
 e-mail: toshi-nakagawa@aitech.ac.jp; mztm@aitech.ac.jp

X. Zhao
 College of Economics and Management, Nanjing University of
 Aeronautics and Astronautics, Nanjing, China
 e-mail: xz.cem@nuaa.edu.cn

like new at failures [2, p. 69]. On the other hand, when the unit undergoes imperfect repair at failures, this is called imperfect maintenance [2, p. 171], and its age becomes younger at repair. Especially, When the unit has the same age before repair at failures, this is called minimal repair [1, p. 96], [2, p. 95] and is commonly used with complex systems such as computers and airplanes. Such two replacement policies have been generally taken up for most reliability models theoretically and practically in maintenance engineering.

It has also been well-known that high system reliability can be achieved through redundancy, e.g., parallel systems with n units in parallel structures [3]. When several replacement policies are planned simultaneously as alternatives, they can be considered as redundancies with each other, such as bivariate policies of replacement first and last [4], trivariate policies of replacement middle [5], and general policies with multiple replacement plans [6].

In this chapter, we name the replacements with redundancies as redundant replacement policies and survey the recent redundant policies that are based on the approaches of whichever occurs first and last, overtime approach for replacement delay, and triggering policies at middle times. From such a viewpoint, we firstly give the single age replacement policies that are planned at time T , at cycle N of working times, and at the first completion of working cycles over time T , respectively [7]. For a large and complex system, which consists of many kinds of units, minimal repairs that cost less are always taken into considerations at failures [1]. Next, we use the cumulative hazard function $H(t)$ to count the number of minimal repairs and survey periodic replacement policies that are planned at time T , at number K of failures, at cycle N of working times, and at the first failure or the first completion of working cycles over time T , respectively.

When a bivariate replacement policy is planned, it has been shown that the policies based on the approaches of whichever occurs first and last [4] become good alternatives from the points of cost and maintainability. For example, replacement first could be done for high system reliability but could not complete the necessary working cycles, while replacement last could let the system operate for a longer time and avoid operational interruptions to complete the running working cycles. On the other hand, when the system is running some successive jobs without stops, it is better to perform replacement policies after several jobs are completed even though the replacement time has arrived [7]. Replacement policies scheduled at the first completion of some working cycle over a planned time T were summarized [8]. That is, replacement scheduled at continuous times could be modified to be done at discrete applications. For the above viewpoint, we combine the above respective single replacement policies for bivariate replacement policies, such that the unit is replaced at time T or at cycle N , whichever occurs first and last, and the unit is replaced at cycle N or at the first

completion of working cycles over time T . When minimal repair is taken into considerations, the number K of failures is modeled for the periodic policies of replacement first and last and replacement overtime first and last [9]. In addition, we begin to plan replacement time T once the K th failure has occurred and plan failure number K once time T has arrived, as the extensions of the above overtime replacement.

When a trivariate replacement policy is planned, i.e., time T , cycle N of working times, and number K of failures are planned simultaneously, we survey the models of replacement first and last, and meanwhile, replacement middle in which the policy is only available at middle times rather than the first or last times [5]. The modified replacement first and last for trivariate policies also contains some cases when replacement is done at middle times. Finally, redundant replacement policies, which means replacement planned at time T , have n redundant policies that can be done at n random times [6]. Age and periodic replacement policies, using the approaches of first and last, are surveyed. For each policy, we give the models of the expected cost rates, and their optimal replacement solutions are obtained analytically and discussed numerically.

Throughout this chapter, it is assumed that the failure time of the unit has a failure distribution $F(t)$, its density function $f(t) \equiv dF(t)/dt$, i.e., $F(t) = \int_0^t f(u)du$ for $t \geq 0$, and its finite mean $\mu \equiv \int_0^\infty \bar{F}(t)dt$, where $\bar{F}(t) \equiv 1 - F(t)$. Furthermore, it is assumed that the failure function is $h(t) \equiv f(t)/\bar{F}(t)$, which increases strictly with t from $h(0) = 0$ to $h(\infty) = \infty$, and the cumulative hazard function is $H(t) \equiv \int_0^t h(u)du$, i.e., $h(t) = dH(t)/dt$ and $F(t) = 1 - e^{-H(t)}$. Generally, we use the following notations throughout this chapter for the simplicity of equations: $\bar{\Phi}(t) \equiv 1 - \Phi(t)$, $\lim_{t \rightarrow 0} \Phi(t) \equiv \Phi(0)$ and $\lim_{t \rightarrow \infty} \Phi(t) \equiv \Phi(\infty)$ for a general function $\Phi(t)$.

When failures occur at a nonhomogeneous Poisson process with hazard rate $H(t)$, the probability that k failures occur in $[0, t]$ is $p_k(t) \equiv [H(t)^k/k!]e^{-H(t)}$ ($k = 0, 1, 2, \dots$), and $P_K(t) \equiv \sum_{k=K}^\infty p_k(t)$ ($K = 0, 1, 2, \dots$), $\bar{P}_K(t) \equiv 1 - P_K(t) = \sum_{k=0}^{K-1} p_k(t)$, where note that $P_K(0) = 0$, $P_K(\infty) = 1$, $P_0(t) = 1$, and $\lim_{K \rightarrow \infty} P_K(t) = 0$. Furthermore, we have the following relations: For $0 < t < \infty$ and $K = 0, 1, 2, \dots$,

$$\begin{aligned}
 P_{K+1}(t) &= \int_0^t p_K(u)h(u)du, \\
 \bar{P}_{K+1}(t) &= \int_t^\infty p_K(u)h(u)du, \\
 \int_0^\infty p_K(t)h(t)dt &= 1, \\
 \int_0^\infty H(t)dP_K(t) &= \int_0^\infty \bar{P}_K(t)h(t)dt \\
 &= \sum_{k=0}^{K-1} \int_0^\infty p_k(t)h(t)dt = K,
 \end{aligned}$$

and letting $N(t)$ be the expected number of failures during $[0, t]$,

$$E\{N(t)\} = \sum_{k=1}^{\infty} k p_k(t) = \sum_{k=1}^{\infty} \bar{P}_k(t) = H(t).$$

In addition, when the unit operates for successive jobs without stops, the working cycles Y_j ($j = 1, 2, \dots$) are independent random variables with an identical distribution $G(t) \equiv \Pr\{Y_j \leq t\}$ with finite mean $1/\theta \equiv \int_0^{\infty} \bar{G}(t) dt$, where $Y_0 \equiv 0$. Furthermore, denote $G^{(j)}(t)$ ($j = 0, 1, 2, \dots$) as the j -fold Stieltjes convolution of $G(t)$ with itself, where $G^{(0)}(t) \equiv 1$ for $t \geq 0$.

28.2 Single Policy

28.2.1 Age Replacement

We introduce the age replacement policies in which the unit is replaced at failure, time T , or cycle N . The expected cost rates are obtained, and optimal policies which minimize them are theoretically shown.

(1) Time T

Suppose that the unit is replaced at a planned time T ($0 < T \leq \infty$) or at failure, whichever occurs first, which is called *standard age replacement*. Then, the expected cost rate is [1, p. 88], [2, p. 72]

$$C(T) = \frac{c_T + (c_F - c_T)F(T)}{\int_0^T \bar{F}(t) dt}, \tag{28.1}$$

where c_T = replacement cost at time T and c_F = replacement cost at failure with $c_F > c_T$. We find optimal T^* to minimize $C(T)$. Differentiating $C(T)$ with respect to T and setting it equal to zero,

$$h(T) \int_0^T \bar{F}(t) dt - F(T) = \frac{c_T}{c_F - c_T}, \tag{28.2}$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T^* ($0 < T^* < \infty$) which satisfies (28.2), and the resulting cost rate is

$$C(T^*) = (c_F - c_T)h(T^*). \tag{28.3}$$

(2) Cycle N

Suppose that the unit is replaced at cycle N ($N = 1, 2, \dots$) of working times or at failure, whichever occurs first, which is called *random replacement*. Then, the expected cost rate is [7, p. 44]

$$C(N) = \frac{c_N + (c_F - c_N) \int_0^{\infty} [1 - G^{(N)}(t)] dF(t)}{\int_0^{\infty} [1 - G^{(N)}(t)] \bar{F}(t) dt}, \tag{28.4}$$

where c_N = replacement cost at cycle N with $c_N < c_F$. We find optimal N^* to minimize $C(N)$. Forming $C(N + 1) - C(N) \geq 0$,

$$Q(N) \int_0^{\infty} [1 - G^{(N)}(t)] \bar{F}(t) dt - \int_0^{\infty} [1 - G^{(N)}(t)] dF(t) \geq \frac{c_N}{c_F - c_N}, \tag{28.5}$$

where

$$Q(N) \equiv \frac{\int_0^{\infty} [G^{(N)}(t) - G^{(N+1)}(t)] dF(t)}{\int_0^{\infty} [G^{(N)}(t) - G^{(N+1)}(t)] \bar{F}(t) dt}.$$

Thus, if $Q(N)$ increases strictly with N to $h(\infty)$, then there exists a finite and unique minimum N^* ($1 \leq N^* < \infty$) which satisfies (28.5).

In particular, when $G(t) = 1 - e^{-\theta t}$, i.e., $G^{(j)}(t) = \sum_{i=j}^{\infty} [(\theta t)^i / i!] e^{-\theta t}$ ($j = 0, 1, 2, \dots$), (28.5) becomes

$$Q(N) \int_0^{\infty} [1 - G^{(N)}(t)] \bar{F}(t) dt - \int_0^{\infty} [1 - G^{(N)}(t)] dF(t) \geq \frac{c_N}{c_F - c_N}, \tag{28.6}$$

where

$$Q(N) \equiv \frac{\int_0^{\infty} (\theta t)^N e^{-\theta t} dF(t)}{\int_0^{\infty} (\theta t)^N e^{-\theta t} \bar{F}(t) dt},$$

which increases strictly with N to $h(\infty)$ from Appendix 2. Thus, noting that the left-hand side of (28.6) increases strictly with N to ∞ , there exists a finite and unique minimum N^* ($1 \leq N^* < \infty$) which satisfies (28.6).

(3) Overtime T

Suppose that the unit is replaced at the first completion of working cycles over time T ($0 \leq T < \infty$) or at failure, whichever occurs first, which is called *age replacement overtime*. Then, the expected cost rate is [7, p. 35], [8, p. 7]

$$C_O(T) = \frac{c_O + (c_F - c_O) \left(F(T) + \sum_{j=0}^{\infty} \int_0^T \left\{ \int_{T-t}^{\infty} [F(t+u) F(T)] dG(u) \right\} dG^{(j)}(t) \right)}{\int_0^T \bar{F}(t) dt + \sum_{j=0}^{\infty} \int_0^T \left[\int_{T-t}^{\infty} \bar{G}(u) \bar{F}(t+u) du \right] \times dG^{(j)}(t)}, \tag{28.7}$$

where c_O = replacement cost over time T with $c_O < c_F$.

In particular, when $G(t) = 1 - e^{-\theta t}$, (28.7) becomes

$$C_O(T) = \frac{c_O + (c_F - c_O) [F(T) + \int_T^{\infty} e^{-\theta(t-T)} dF(t)]}{\int_0^T \bar{F}(t) dt + \int_T^{\infty} e^{-\theta(t-T)} \bar{F}(t) dt}. \tag{28.8}$$

We find optimal T_O^* to minimize $C_O(T)$. Differentiating $C_O(T)$ with respect to T and setting it equal to zero,

$$\tilde{Q}_O(T) \int_0^T \bar{F}(t) dt - F(T) = \frac{c_O}{c_F - c_O}, \tag{28.9}$$

where

$$\tilde{Q}_O(T) \equiv \frac{\int_T^\infty e^{-\theta t} dF(t)}{\int_T^\infty e^{-\theta t} \bar{F}(t) dt},$$

which increases strictly with T to $h(\infty)$ from Appendix 1. Thus, noting that the left-hand side of (28.9) increases strictly with T from 0 to ∞ , there exists a finite and unique T_O^* ($0 < T_O^* < \infty$) which satisfies (28.9), and the resulting cost rate is

$$\begin{aligned} C_O(T_O^*) &= (c_F - c_O) \tilde{Q}_O(T_O^*) \\ &= \frac{c_F - (c_F - c_O) \bar{F}(T_O^*)}{\int_0^{T_O^*} \bar{F}(t) dt}, \end{aligned} \tag{28.10}$$

which agrees with (28.1) when $c_O = c_T$ and $T_O^* = T$. In addition, when $c_O = c_T$, comparing (28.9) with (28.2), $T_O^* < T^*$ because $\tilde{Q}_O(T) > h(T)$. Therefore, from (28.10), $C_O(T_O^*) > C(T^*)$, i.e., age replacement is better than replacement overtime when both replacement costs are the same.

28.2.2 Periodic Replacement

An operating unit undergoes minimal repair at each failure, and the failure rate remains undisturbed by any repair of failures between replacements. Then, we introduce the standard periodic replacement policies in which the unit is replaced at time T , failure K , and cycle N . The results obtained in this section could be applied easily to imperfect maintenance policies [2, p. 175].

(1) Time T

Suppose that the unit is replaced at a planned time T ($0 < T < \infty$) which is called *standard periodic replacement*. Then, the expected cost rate is [1, p. 97], [2, p. 102]

$$C(T) = \frac{c_T + c_M H(T)}{T}, \tag{28.11}$$

where c_T = replacement cost at time T and c_M = cost of minimal repair at each failure. We find optimal T^* to minimize $C(T)$. Differentiating $C(T)$ with respect to T and setting it equal to zero,

$$Th(T) - H(T) = \frac{c_T}{c_M}, \tag{28.12}$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T^* ($0 < T^* < \infty$) which satisfies (28.12), and the resulting cost rate is

$$C(T^*) = c_M h(T^*). \tag{28.13}$$

(2) Failure K

Suppose that the unit is replaced at a number K ($K = 1, 2, \dots$) of failures. Then, the expected cost rate is [2, p. 104]

$$C(K) = \frac{c_K + c_M K}{\int_0^\infty \bar{P}_K(t) dt}, \tag{28.14}$$

where c_K = replacement cost at failure K . We find optimal K^* to minimize $C(K)$. Forming $C(K + 1) - C(K) \geq 0$,

$$\frac{1}{\int_0^\infty \bar{P}_K(t) dt} - K \geq \frac{c_K}{c_M}, \tag{28.15}$$

whose left-hand increases strictly with K to ∞ . Thus, there exists a finite and unique minimum K^* ($1 \leq K^* < \infty$) which satisfies (28.15).

(3) Cycle N

Suppose that the unit is replaced at cycle N ($N = 1, 2, \dots$) of working times. Then, the expected cost rate is [7, p. 76]

$$C(N) = \frac{c_N + c_M \int_0^\infty [1 - G^{(N)}(t)] h(t) dt}{N/\theta}, \tag{28.16}$$

where c_N = replacement cost at cycle N . We find optimal N^* to minimize $C(N)$. Forming $C(N + 1) - C(N) \geq 0$,

$$\frac{NH(N)}{\theta} - \int_0^\infty [1 - G^{(N)}(t)] h(t) dt \geq \frac{c_N}{c_M}, \tag{28.17}$$

where

$$H(N) \equiv \theta \int_0^\infty [G^{(N)}(t) - G^{(N+1)}(t)] h(t) dt.$$

Thus, if $H(N)$ increases strictly with N to $h(\infty)$, then noting that the left-hand side of (28.17) increases strictly with N to ∞ , there exists a finite and unique minimum N^* ($1 \leq N^* < \infty$) which satisfies (28.17).

In particular, when $G(t) = 1 - e^{-\theta t}$, (28.17) becomes

$$\frac{NH(N)}{\theta} - \int_0^\infty [1 - G^{(N)}(t)] h(t) dt \geq \frac{c_N}{c_M}, \tag{28.18}$$

where

$$H(N) \equiv \int_0^\infty \frac{\theta(\theta t)^N}{N!} e^{-\theta t} h(t) dt,$$

which increases strictly with N to $h(\infty)$ from Appendix 3. Thus, there exists a finite and unique minimum N^* ($1 \leq N^* < \infty$) which satisfies (28.18).

(4) Overtime T

Suppose that the unit is replaced at the first failure over time T , which is called *periodic replacement overtime*. Then, the expected cost rate is [8, p. 47]

$$C_{OK}(T) = \frac{c_{OK} + c_M[H(T) + 1]}{T + \int_T^\infty e^{-H(t)+H(T)}dt}, \quad (28.19)$$

where c_{OK} = replacement cost at the first failure over time T . We find optimal T_{OK}^* to minimize $C_{OK}(T)$. Differentiating $C_{OK}(T)$ with respect to T and setting it equal to zero,

$$TQ(T) - H(T) = \frac{c_{OK}}{c_M}, \quad (28.20)$$

where

$$Q(T) \equiv \frac{1}{\int_T^\infty e^{-H(t)+H(T)}dt} = \frac{\bar{F}(T)}{\int_T^\infty \bar{F}(t)dt},$$

which increases strictly with T from $1/\mu$ to $h(\infty)$ from Appendix 1. Thus, noting that the left-hand side of (28.20) increases strictly with T from 0 to ∞ , there exists a finite and unique T_{OK}^* ($0 < T_{OK}^* < \infty$) which satisfies (28.20), and the resulting cost rate is

$$C_{OK}(T_{OK}^*) = c_{OK}Q(T_{OK}^*) = \frac{c_{OK} + c_M H(T_{OK}^*)}{T_{OK}^*}, \quad (28.21)$$

which agrees with (28.11) when $c_{OK} = c_T$ and $T_{OK}^* = T$. In addition, when $c_{OK} = c_T$, comparing (28.20) with (28.12), $T_{OK}^* < T^*$, because $Q(T) > h(T)$. Therefore, from (28.21), $C_{OK}(T_{OK}^*) > C(T^*)$, i.e., periodic replacement is better than replacement overtime when both replacement costs are the same.

Next, suppose that the unit is replaced at the first completion of working cycles over time T . Then, the expected cost rate is [8, p. 39]

$$C_{ON}(T) = \frac{c_{ON} + c_M \sum_{j=0}^\infty \int_0^T [\int_0^\infty \bar{G}(u)h(t+u)du]dG^{(j)}(t)}{(1/\theta) \sum_{j=0}^\infty G^{(j)}(T)}, \quad (28.22)$$

where c_{ON} = replacement cost at the first cycle over time T . We find optimal T_{ON}^* to minimize $C_{ON}(T)$. Differentiating $C_{ON}(T)$ with respect to T and setting it equal to zero,

$$\int_0^\infty \bar{G}(t)h(T+t)dt \sum_{j=0}^\infty G^{(j)}(T) - \sum_{j=0}^\infty \int_0^T \left[\int_0^\infty \bar{G}(u)h(t+u)du \right]dG^{(j)}(t) = \frac{c_{ON}}{c_M}, \quad (28.23)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_{ON}^* ($0 < T_{ON}^* < \infty$) which satisfies (28.23).

In particular, when $G(t) = 1 - e^{-\theta t}$, (28.23) becomes

$$T \int_0^\infty \theta e^{-\theta t}h(T+t)dt - H(T) = \frac{c_{ON}}{c_M}, \quad (28.24)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_{ON}^* ($0 < T_{ON}^* < \infty$) which satisfies (28.24), and the resulting cost rate is

$$C_{ON}(T_{ON}^*) = c_M \int_0^\infty \theta e^{-\theta t}h(T_{ON}^*+t)dt = \frac{c_{ON} + c_M H(T_{ON}^*)}{T_{ON}^*}, \quad (28.25)$$

which agrees with (28.11) when $c_{ON} = c_T$ and $T_{ON}^* = T$. In addition, when $c_{ON} = c_T$, comparing (28.24) with (28.12), $T_{ON}^* < T^*$, because $\int_0^\infty \theta e^{-\theta t}h(T+t)dt > h(T)$. Therefore, from (28.25), $C_{ON}(T_{ON}^*) > C(T^*)$, i.e., periodic replacement is better than replacement overtime when both replacement costs are the same. However, if replacement cost at overtime would be smaller than that at time T , then replacement overtime might be better than periodic replacement.

28.3 Two Policies

28.3.1 Age Replacement

Combining age replacements with time T and cycle N in Sect. 28.2.1, we introduce the following four policies:

(1) Replacement First

Suppose that the unit is replaced preventively at time T ($0 < T \leq \infty$) or at cycle N ($N = 1, 2, \dots$), whichever occurs first. Then, the expected cost rate is [7, p. 43]

$$C_F(T, N) = \frac{c_T + (c_F - c_T) \int_0^T [1 - G^{(N)}(t)]dF(t) + (c_N - c_T) \int_0^T \bar{F}(t)dG^{(N)}(t)}{\int_0^T [1 - G^{(N)}(t)]\bar{F}(t)dt}. \quad (28.26)$$

Note that $C_F(T, \infty) = C(T)$ in (28.1) and $C_F(\infty, N) = C(N)$ in (28.4).

When $c_T = c_N$, we find optimal T_F^* and N_F^* to minimize $C_F(T, N)$. Differentiating $C_F(T, N)$ with respect to T and setting it equal to zero,

$$h(T) \int_0^T [1 - G^{(N)}(t)]\bar{F}(t)dt - \int_0^T [1 - G^{(N)}(t)]dF(t) = \frac{c_T}{c_F - c_T}, \quad (28.27)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_F^* ($0 < T_F^* < \infty$) which satisfies (28.27) for any N , and the resulting cost rate is

$$C_F(T_F^*, N) = (c_F - c_T)h(T_F^*). \quad (28.28)$$

Furthermore, noting that the left-hand side of (28.27) increases with N to that of (28.2), T_F^* decreases with N to T^* given in (28.2), and $T_F^* \geq T^*$.

Forming $C_F(T, N + 1) - C_F(T, N) \geq 0$,

$$Q(T, N) \int_0^T [1 - G^{(N)}(t)]\bar{F}(t)dt - \int_0^T [1 - G^{(N)}(t)]dF(t) \geq \frac{c_T}{c_F - c_T}, \quad (28.29)$$

where

$$Q(T, N) \equiv \frac{\int_0^T [G^{(N)}(t) - G^{(N+1)}(t)]dF(t)}{\int_0^T [G^{(N)}(t) - G^{(N+1)}(t)]\bar{F}(t)dt},$$

and $Q(T, N) < h(T)$. Substituting (28.27) for (28.29),

$$Q(T, N) \geq h(T),$$

which does not hold for any T , i.e., $C_F(T, N + 1) < C_F(T, N)$ and $N_F^* = \infty$. Therefore, the optimal policy which minimizes $C_F(T, N)$ is $N_F^* = \infty$ and $T_F^* = T^*$ given in (28.2).

In particular, when $G(t) = 1 - e^{-\theta t}$,

$$Q(T, N) = \frac{\int_0^T (\theta t)^N e^{-\theta t} dF(t)}{\int_0^T (\theta t)^N e^{-\theta t} \bar{F}(t) dt},$$

which increases strictly with T to $Q(N)$ in (28.6) and increases strictly with N to $h(T)$ from Appendix 2. Thus, the left-hand side of (28.29) increases strictly with N to that of (28.2). Therefore, if $T \leq T^*$, then $N_F^* = \infty$, and conversely, if $T > T^*$, then there exists a finite and unique minimum N_F^* ($1 \leq N_F^* < \infty$) which satisfies (28.29). In addition, because the left-hand side of (28.29) increases with T to that of (28.6), N_F^* decreases with T to N^* given in (28.6), and $N_F^* \geq N^*$.

(2) Replacement Last

Suppose that the unit is replaced preventively at time T ($0 \leq T < \infty$) or at cycle N ($N = 0, 1, 2, \dots$), whichever occurs last. Then, the expected cost rate is [7, p. 47]

$$C_L(T, N) = \frac{c_T + (c_F - c_T) \left\{ F(T) + \int_T^\infty [1 - G^{(N)}(t)] dF(t) \right\} + (c_N - c_T) \int_T^\infty \bar{F}(t) dG^{(N)}(t)}{\int_0^T \bar{F}(t) dt + \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt}, \quad (28.30)$$

which agrees with (28.1) when $N = 0$ and (28.4) when $T = 0$.

When $c_T = c_N$, we find optimal T_L^* and N_L^* to minimize $C_L(T, N)$. Differentiating $C_L(T, N)$ with respect to T and setting it equal to zero,

$$h(T) \left\{ \int_0^T \bar{F}(t) dt + \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \right\} - F(T) - \int_T^\infty [1 - G^{(N)}(t)] dF(t) = \frac{c_T}{c_F - c_T}, \quad (28.31)$$

whose left-hand side increases strictly to ∞ . Thus, there exists a finite and unique T_L^* ($0 \leq T_L^* < \infty$) which satisfies (28.31) for any N , and the resulting cost rate is

$$C_L(T_L^*, N) = (c_F - c_T)h(T_L^*). \quad (28.32)$$

Furthermore, noting that the left-hand side of (28.31) decreases with N from that of (28.2), T_L^* increases with N from T^* given in (28.2), and $T_L^* \geq T^*$.

Forming $C_L(T, N + 1) - C_L(T, N) \geq 0$,

$$\tilde{Q}(T, N) \left\{ \int_0^T \bar{F}(t) dt + \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \right\} - F(T) - \int_T^\infty [1 - G^{(N)}(t)] dF(t) \geq \frac{c_T}{c_F - c_T}, \quad (28.33)$$

where

$$\tilde{Q}(T, N) = \frac{\int_T^\infty [G^{(N)}(t) - G^{(N+1)}(t)] dF(t)}{\int_T^\infty [G^{(N)}(t) - G^{(N+1)}(t)] \bar{F}(t) dt},$$

and $\tilde{Q}(T, N) > h(T)$. Substituting (28.31) for (28.33),

$$\tilde{Q}(T, N) \geq h(T),$$

which always holds for any T , i.e., $C_L(T, N + 1) \geq C_L(T, N)$. Therefore, the optimal policy which minimizes $C_L(T, N)$ is $N_L^* = 0$ and $T_L^* = T^*$ given in (28.2).

In particular, when $G(t) = 1 - e^{-\theta t}$,

$$\tilde{Q}(T, N) = \frac{\int_T^\infty (\theta t)^N e^{-\theta t} dF(t)}{\int_T^\infty (\theta t)^N e^{-\theta t} \bar{F}(t) dt},$$

which increases strictly with T from $Q(N)$ to $h(\infty)$ and increases strictly with N from $\tilde{Q}_0(T)$ to $h(\infty)$ from Appendix 2. Thus, the left-hand side of (28.33) increases strictly with N from

$$\tilde{Q}_0(T) \int_0^T \bar{F}(t) dt - F(T) > h(T) \int_0^T \bar{F}(t) dt - F(T),$$

which agrees with (28.2). Therefore, if $T \geq T^*$, then $N_L^* = 0$, and conversely, if $T < T^*$, then there exists a finite and unique minimum N_L^* ($0 \leq N_L^* < \infty$) which satisfies (28.33). In addition, the left-hand side of (28.33) increases with T from that of (28.6), N_L^* decreases with T from N^* given in (28.6), and $N_L^* \leq N^*$.

Therefore, from the above results, if $T > T^*$, replacement first is better than replacement last, if $T = T^*$, then age replacement is better than replacement first and last, and if $T < T^*$, replacement last is better than replacement first.

Finally, compare replacement first and replacement last for given N . From (28.27) and (28.31), denoting

$$\begin{aligned}
 L_1(T; N) &\equiv \\
 &h(T) \left\{ \int_0^T \bar{F}(t)dt + \int_T^\infty [1 - G^{(N)}(t)]\bar{F}(t)dt \right\} \\
 &\quad - F(T) - \int_T^\infty [1 - G^{(N)}(t)]dF(t) \\
 &\quad - h(T) \int_0^T [1 - G^{(N)}(t)]\bar{F}(t)dt \\
 &\quad + \int_0^T [1 - G^{(N)}(t)]dF(t) \\
 &= \int_0^T G^{(N)}(t)\bar{F}(t)[h(T) - h(t)]dt \\
 &\quad - \int_T^\infty [1 - G^{(N)}(t)]\bar{F}(t)[h(t) - h(T)]dt,
 \end{aligned}$$

we have

$$\begin{aligned}
 L_1(0; N) &= - \int_0^\infty [1 - G^{(N)}(t)]dF(t) < 0, \\
 L_1(\infty) &= \infty, \\
 L_1'(T; N) &= h'(T) \left\{ \int_0^T G^{(N)}(t)\bar{F}(t)dt \right. \\
 &\quad \left. + \int_T^\infty [1 - G^{(N)}(t)]\bar{F}(t)dt \right\} > 0.
 \end{aligned}$$

Thus, there exists a finite and unique T_A ($0 < T_A < \infty$) which satisfies $L_1(T; N) = 0$. In addition, noting that $L_1(T; N)$ decreases with N , T_A increases with N from T^* given in (28.2) to ∞ . We set that

$$\begin{aligned}
 L(T_A) &\equiv h(T_A) \int_0^{T_A} [1 - G^{(N)}(t)]\bar{F}(t)dt \\
 &\quad - \int_0^{T_A} [1 - G^{(N)}(t)]dF(t).
 \end{aligned}$$

Then, it is shown from (28.27) and (28.31) that if $L(T_A) \geq c_T/(c_F - c_T)$, i.e., $c_F/c_T \geq 1 + 1/L(T_A)$, then $T_F^* \leq T_L^*$, and from (28.28) and (28.32), replacement first is better than replacement last. Conversely, if $L(T_A) < c_T/(c_F - c_T)$, then $T_F^* > T_L^*$, and replacement last is better than replacement first. This means that if the ratio of c_T/c_F is greater than $L(T_A)/[1 + L(T_A)]$, i.e., replacement cost c_F is nearly equal to cost c_T , replacement last is better than replacement first.

On the other hand, noting that $L_1(T, N)$ decreases with N from

$$L_1(T; 0) = \int_0^T \bar{F}(t)[h(T) - h(t)]dt > 0,$$

to

$$L_1(T; \infty) = - \int_T^\infty \bar{F}(t)[h(t) - h(T)]dt < 0,$$

there exists a finite and unique maximum N_A ($0 \leq N_A < \infty$) which satisfies $L_1(T, N) > 0$. Thus, if $N \geq N_A$, then $L_1(T, N) \leq 0$, i.e., $T_F^* \leq T_L^*$ and replacement first is better than replacement last. Conversely, if $N < N_A$, then $L_1(T, N) > 0$, i.e., $T_F^* > T_L^*$ and replacement last is better than replacement first.

From the above results, if c_F and N are small, replacement last is better than replacement first, and vice versa.

(3) Replacement Overtime First

Suppose that the unit is replaced preventively at cycle N ($N = 1, 2, \dots$) before time T or at the first completion of working cycles over time T ($0 \leq T < \infty$), whichever occurs first. Then, the expected cost rate is [8, p. 9]

$$\begin{aligned}
 C_{OF}(T, N) &= \\
 &\frac{c_O + (c_N - c_O) \int_0^T \bar{F}(t)dG^{(N)}(t) + (c_F - c_O) \left(\int_0^T [1 - G^{(N)}(t)]dF(t) + \sum_{j=0}^{N-1} \int_0^T \left\{ \int_{T-t}^\infty [F(t+u) - F(T)]dG^{(j)}(u) \right\} \times dG^{(j)}(t) \right)}{\int_0^T [1 - G^{(N)}(t)]\bar{F}(t)dt + \sum_{j=0}^{N-1} \int_0^T \left[\int_{T-t}^\infty \bar{G}(u)\bar{F}(t+u)du \right]dG^{(j)}(t)},
 \end{aligned} \tag{28.34}$$

which agrees with (28.4) when $T = \infty$ and (28.7) when $N = \infty$.

When $c_O = c_N$ and $G(t) = 1 - e^{-\theta t}$, the expected cost rate is

$$\begin{aligned}
 C_{OF}(T, N) &= \\
 &\frac{c_O + (c_F - c_O) \left\{ \int_0^T [1 - G^{(N)}(t)]dF(t) + [1 - G^{(N)}(T)] \int_T^\infty e^{-\theta(t-T)}dF(t) \right\}}{\int_0^T [1 - G^{(N)}(t)]\bar{F}(t)dt + [1 - G^{(N)}(T)] \int_T^\infty e^{-\theta(t-T)}\bar{F}(t)dt}.
 \end{aligned} \tag{28.35}$$

We find optimal T_{OF}^* and N_{OF}^* to minimize $C_{OF}(T, N)$. Differentiating $C_{OF}(T, N)$ with respect to T and setting it equal to zero,

$$\begin{aligned} &\tilde{Q}_O(T) \int_0^T [1 - G^{(N)}(t)] \bar{F}(t) dt \\ &- \int_0^T [1 - G^{(N)}(t)] dF(t) = \frac{c_O}{c_F - c_O}, \end{aligned} \quad (28.36)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_{OF}^* ($0 < T_{OF}^* < \infty$) which satisfies (28.36) for any N , and the resulting cost rate is

$$\begin{aligned} C_{OF}(T_{OF}^*) &= (c_F - c_O) \tilde{Q}_O(T_{OF}^*) \\ &= \frac{c_O + (c_F - c_O) \int_0^{T_{OF}^*} [1 - G^{(N)}(t)] dF(t)}{\int_0^{T_{OF}^*} [1 - G^{(N)}(t)] \bar{F}(t) dt}, \end{aligned} \quad (28.37)$$

which agrees with (28.26) when $c_O = c_T = c_N$ and $T_{OF}^* = T$. Furthermore, noting that the left-hand side of (28.36) increases strictly with N to that of (28.9), T_{OF}^* decreases with N to T_O^* given in (28.9), and $T_{OF}^* \geq T_O^*$.

Forming $C_{OF}(T, N + 1) - C_{OF}(T, N) \geq 0$,

$$\begin{aligned} &Q_1(T, N - 1) \left\{ [1 - G^{(N)}(T)] \int_T^\infty e^{-\theta(t-T)} \bar{F}(t) dt \right. \\ &+ \left. \int_0^T [1 - G^{(N)}(t)] \bar{F}(t) dt \right\} \\ &- [1 - G^{(N)}(T)] \int_T^\infty e^{-\theta(t-T)} dF(t) \\ &- \int_0^T [1 - G^{(N)}(t)] dF(t) \geq \frac{c_O}{c_F - c_O}, \end{aligned} \quad (28.38)$$

where

$$Q_1(T, N) \equiv \frac{\int_0^T (\theta t)^N \left[\int_t^\infty e^{-\theta u} dF(u) \right] dt}{\int_0^T (\theta t)^N \left[\int_t^\infty e^{-\theta u} \bar{F}(u) du \right] dt},$$

which increases strictly with T to $Q_1(\infty, N) = Q(N + 1)$, and increases strictly with N to $\tilde{Q}_O(T)$, and $Q_1(T, N) < \tilde{Q}_O(T)$ from Appendix 4. Substituting (28.36) for (28.38),

$$\begin{aligned} &Q_1(T, N - 1) \left\{ [1 - G^{(N)}(T)] \int_T^\infty e^{-\theta(t-T)} \bar{F}(t) dt \right. \\ &+ \left. \int_0^T [1 - G^{(N)}(t)] \bar{F}(t) dt \right\} \\ &- [1 - G^{(N)}(T)] \int_T^\infty e^{-\theta(t-T)} dF(t) \\ &\geq \tilde{Q}_O(T) \int_0^T [1 - G^{(N)}(t)] \bar{F}(t) dt, \end{aligned}$$

which does not hold for any T , i.e., $C_{OF}(T, N + 1) \leq C_{OF}(T, N)$. Therefore, the optimal policy which minimizes $C_{OF}(T, N)$ is $N_{OF}^* = \infty$ and $T_{OF}^* = T_O^*$ given in (28.9).

Furthermore, note that the left-hand side of (28.38) increases strictly with N to that of (28.9). Therefore, if $T \leq T_O^*$, then $N_{OF}^* = \infty$, and conversely, if $T > T_O^*$, then there exists a finite and unique minimum N_{OF}^* ($1 \leq N_{OF}^* < \infty$) which satisfies (28.38). In addition, because the left-hand side of (28.38) increases with T to

$$\begin{aligned} &Q_1(\infty, N - 1) \int_0^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \\ &- \int_0^\infty [1 - G^{(N)}(t)] dF(t) < \\ &Q(N) \int_0^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \\ &- \int_0^\infty [1 - G^{(N)}(t)] dF(t), \end{aligned}$$

which agrees with (28.6), N_{OF}^* decreases with T to N^* given in (28.6), and $N_{OF}^* \geq N^*$.

(4) Replacement Overtime Last

Suppose that the unit is preventively at cycle N ($N = 0, 1, 2, \dots$) or at the first completion of working cycles over time T ($0 \leq T < \infty$), whichever occurs last. Then, the expected cost rate is [8, p. 13]

$$\begin{aligned} C_{OL}(T, N) &= \\ &\frac{c_O + (c_N - c_O) \int_T^\infty \bar{F}(t) dG^{(N)}(t) + (c_F - c_O) (F(T) + \int_T^\infty [F(t) - F(T)] dG^{(N)}(t) + \sum_{j=N}^\infty \int_0^T \{ \int_{T-t}^\infty [F(t+u) - F(T)] dG^{(j)}(u) \} dG^{(j)}(t))}{\int_0^T \bar{F}(t) dt + \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt + \sum_{j=N}^\infty \int_0^T [\int_{T-t}^\infty \bar{G}(u) \bar{F}(t+u) du] dG^{(j)}(t)}, \end{aligned} \quad (28.39)$$

which agrees with (28.4) when $T = 0$ and (28.7) when $N = 0$, and note that $C_{OL}(T, 0) = C_{OL}(T, 1)$.

When $c_O = c_N$ and $G(t) = 1 - e^{-\theta t}$, the expected cost rate is

$$\begin{aligned} C_{OL}(T, N) &= \\ &\frac{c_O + (c_F - c_O) \{ F(T) + \int_T^\infty [1 - G^{(N)}(t)] dF(t) + G^{(N)}(T) \int_T^\infty e^{-\theta(t-T)} dF(t) \}}{\int_0^T \bar{F}(t) dt + \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt + G^{(N)}(T) \int_T^\infty e^{-\theta(t-T)} \bar{F}(t) dt}. \end{aligned} \quad (28.40)$$

We find optimal T_{OL}^* and N_{OL}^* to minimize $C_{OL}(T, N)$. Differentiating $C_{OL}(T, N)$ with respect to T and setting it equal to zero,

$$\begin{aligned} & \tilde{Q}_O(T) \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \\ & - \int_T^\infty [1 - G^{(N)}(t)] dF(t) \\ & + \tilde{Q}_O(T) \int_0^T \bar{F}(t) dt - F(T) = \frac{c_O}{c_F - c_O}, \quad (28.41) \end{aligned}$$

$$\begin{aligned} & + \tilde{Q}_1(T, N - 1) \int_0^T \bar{F}(t) dt \\ & \geq \tilde{Q}_O(T) \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \\ & + \tilde{Q}_O(T) \int_0^T \bar{F}(t) dt, \end{aligned}$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_{OL}^* ($0 < T_{OL}^* < \infty$) which satisfies (28.41) for any N , and the resulting cost rate is

$$\begin{aligned} C_{OL}(T_{OL}^*, N) &= (c_F - c_O) \tilde{Q}_O(T_{OL}^*) = \\ & \frac{c_O + (c_F - c_O) \{ F(T_{OL}^*) + \int_{T_{OL}^*}^\infty [1 - G^{(N)}(t)] dF(t) \}}{\int_0^{T_{OL}^*} \bar{F}(t) dt + \int_{T_{OL}^*}^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt}, \quad (28.42) \end{aligned}$$

which agrees with (28.30) when $c_O = c_T = c_N$ and $T_{OL}^* = T$. Furthermore, noting that the left-hand side of (28.41) decreases with N from that of (28.9), T_{OL}^* increases with N from T_O^* given in (28.9), and $T_{OL}^* \geq T_O^*$.

Forming $C_{OL}(T, N + 1) - C_{OL}(T, N) \geq 0$,

$$\begin{aligned} & \tilde{Q}_1(T, N - 1) \left\{ \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \right. \\ & \left. + G^{(N)}(T) \int_T^\infty e^{-\theta(t-T)} \bar{F}(t) dt \right\} \\ & - \int_T^\infty [1 - G^{(N)}(t)] dF(t) \\ & - G^{(N)}(T) \int_T^\infty e^{-\theta(t-T)} dF(t) \\ & + \tilde{Q}_1(T, N - 1) \int_0^T \bar{F}(t) dt - F(T) \geq \frac{c_O}{c_F - c_O}, \quad (28.43) \end{aligned}$$

where

$$\tilde{Q}_1(T, N) \equiv \frac{\int_T^\infty (\theta t)^N \left[\int_t^\infty e^{-\theta u} dF(u) \right] dt}{\int_T^\infty (\theta t)^N \left[\int_t^\infty e^{-\theta u} \bar{F}(u) du \right] dt},$$

which increases strictly with T from $Q(N + 1)$ to $h(\infty)$ and increases strictly with N from $\tilde{Q}_1(T, 0)$ to $h(\infty)$, and $\tilde{Q}_1(T, N) > \tilde{Q}_O(T) > Q_1(T, N)$ from Appendix 4. Substituting (28.41) for (28.43),

$$\begin{aligned} & \tilde{Q}_1(T, N - 1) \left\{ \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \right. \\ & \left. + G^{(N)}(T) \int_T^\infty e^{-\theta(t-T)} \bar{F}(t) dt \right\} \\ & - G^{(N)}(T) \int_T^\infty e^{-\theta(t-T)} dF(t) \end{aligned}$$

which always holds for any T , i.e., $C_{OL}(T, N + 1) \geq C_{OL}(T, N)$. Therefore, the optimal policy which minimizes $C_{OL}(T, N)$ is $N_{OL}^* = 0$ or 1 and $T_{OL}^* = T_O^*$ given in (28.9).

Furthermore, note that the left-hand side of (28.43) increases strictly with N from more than that of (28.9). Therefore, if $T \geq T_O^*$, then $N_{OL}^* = 1$ and conversely, if $T < T_O^*$, then there exists a finite and unique minimum N_{OL}^* ($1 \leq N_{OL}^* < \infty$) which satisfies (28.43). In addition, because the left-hand side of (28.43) decreases with T from

$$Q(N) \int_0^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt - \int_0^\infty [1 - G^{(N)}(t)] dF(t),$$

which agrees with (28.6), N_{OL}^* increases with T from N^* given in (28.6), and $N_{OL}^* \geq N^*$.

Therefore, from the above results, if $T > T_O^*$, replacement overtime first is better than overtime last, if $T = T_O^*$, then replacement overtime is better than overtime first and last, and if $T < T_O^*$, replacement overtime last is better than overtime first.

Finally, compare replacement overtime first and last for given N ($N \geq 1$) when $G(t) = 1 - e^{-\theta t}$. From (28.36) and (28.41), denoting

$$\begin{aligned} L_2(T, N) &\equiv \int_0^T G^{(N)}(t) \bar{F}(t) [\tilde{Q}_O(T) - h(t)] dt \\ &- \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) [h(t) - \tilde{Q}_O(T)] dt, \quad (28.44) \end{aligned}$$

we have

$$\begin{aligned} L_2(0, N) &= \\ & - \int_0^\infty [1 - G^{(N)}(t)] \bar{F}(t) [h(t) - \tilde{Q}_O(0)] dt < 0, \end{aligned}$$

$$L_2(\infty) = \infty,$$

$$\begin{aligned} L_2(T, N) &= \frac{e^{-\theta T} \bar{F}(T) [\tilde{Q}_O(T) - h(T)]}{\int_T^\infty e^{-\theta t} \bar{F}(t) dt} \\ &\times \left\{ \int_0^T G^{(N)}(t) \bar{F}(t) dt + \int_T^\infty [1 - G^{(N)}(t)] \bar{F}(t) dt \right\} \\ &+ \bar{F}(T) [\tilde{Q}_O(T) - h(T)] > 0. \end{aligned}$$

Thus, there exists a finite and unique T_{OA} ($0 < T_{OA} < \infty$) which satisfies $L_2(T, N) \equiv 0$. In addition, noting that $L_2(T, N)$ decreases with N , T_{OA} increases with N from T_O^* . We set that

$$L(T_{OA}) \equiv \int_0^{T_{OA}} [1 - G^{(N)}(t)] \bar{F}(t) [\tilde{Q}_O(T_{OA}) - h(t)] dt.$$

Then, it is shown from (28.36) and (28.41) that if $L(T_{OA}) > c_O/(c_F - c_O)$, i.e., $c_F/c_O \geq 1 + 1/L(T_{OA})$, then $T_{OF}^* \leq T_{OL}^*$, and from (28.37) and (28.42), replacement overtime first is better than overtime last. Conversely, if $L(T_{OA}) < c_O/(c_F - c_O)$, then $T_{OF}^* > T_{OL}^*$, and replacement overtime last is better than overtime first. This means that if the ratio of c_O/c_F is greater than $L(T_{OA})/[1 + L(T_{OA})]$, i.e., replacement cost c_F is nearly equal to cost c_O , replacement overtime last is better than overtime first.

On the other hand, noting that $L_2(T, N)$ decreases with N from

$$L_2(T; 0) = \int_0^T \bar{F}(t) [\tilde{Q}_O(T) - h(t)] dt > 0,$$

to

$$L_2(T; \infty) = - \int_T^\infty \bar{F}(t) [h(t) - \tilde{Q}_O(T)] dt < 0,$$

there exists a finite and unique maximum N_{OA} ($0 \leq N_{OA} < \infty$) which satisfies $L_2(T, N) > 0$. Thus, if $N \geq N_{OA}$, then $L_2(T, N) \leq 0$, i.e., $T_{OF}^* \leq T_{OL}^*$, and replacement overtime first is better than overtime last. Conversely, if $N < N_{OA}$, then $L_2(T, N) > 0$, i.e., $T_{OF}^* > T_{OL}^*$, and replacement overtime last is better than overtime first. From the above results, if c_F and N are small, replacement overtime last is better than overtime first and vice versa.

28.3.2 Periodic Replacement

Combining replacements with time T , failure K , and cycle N in Sect. 28.2.2, we introduce the following twelve policies:

(1) Replacement First with Time T and Failure K

Suppose that the unit is replaced at time T ($0 < T \leq \infty$) or at failure K ($K = 1, 2, \dots$), whichever occurs first. Then, the expected cost rate is [9]

$$C_F(T, K) = \frac{c_T + (c_K - c_T)P_K(T) + c_M \int_0^T \bar{P}_K(t)h(t)dt}{\int_0^T \bar{P}_K(t)dt}, \quad (28.45)$$

which agrees with (28.11) when $K = \infty$ and (28.14) when $T = \infty$.

When $c_T = c_K$, we find optimal T_F^* and K_F^* to minimize $C_F(T, K)$. Differentiating $C_F(T, K)$ with respect to T and setting it equal to zero,

$$h(T) \int_0^T \bar{P}_K(t)dt - \int_0^T \bar{P}_K(t)h(t)dt = \frac{c_T}{c_M}, \quad (28.46)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_F^* ($0 < T_F^* < \infty$) which satisfies (28.46) for any K , and the resulting cost rate is

$$C_F(T_F^*, K) = c_M h(T_F^*). \quad (28.47)$$

Furthermore, noting that the left-hand side of (28.46) increases with K to that of (28.12), T_F^* decreases with K to T^* given in (28.12), and $T_F^* \geq T^*$.

Forming $C_F(T, K+1) - C_F(T, K) \geq 0$,

$$H(T, K) \int_0^T \bar{P}_K(t)dt - \int_0^T \bar{P}_K(t)h(t)dt \geq \frac{c_T}{c_M}, \quad (28.48)$$

where

$$H(T, K) \equiv \frac{\int_0^T p_K(t)h(t)dt}{\int_0^T p_K(t)dt},$$

which increases strictly with T to $H(\infty, K) = 1/\int_0^\infty p_K(t)dt$ and increases strictly with K to $h(T)$, and $H(T, K) < h(T)$ from Appendix 5. Substituting (28.46) for (28.48),

$$H(T, K) \geq h(T),$$

which does not hold for any T , i.e., $C_F(T, K+1) < C_F(T, K)$. Therefore, the optimal policy which minimizes $C_F(T, K)$ is $K_F^* = \infty$ and $T_F^* = T^*$ given in (28.12).

Furthermore, noting that the left-hand side of (28.48) increases strictly with K to that of (28.12), if $T \leq T^*$, then $K_F^* = \infty$, and if $T > T^*$, then there exists a finite and unique minimum K_F^* ($1 \leq K_F^* < \infty$) which satisfies (28.48). In addition, because the left-hand side of (28.48) increases with T to that of (28.15), K_F^* decreases with T to K^* given in (28.15), and $K_F^* \geq K^*$.

(2) Replacement Last with Time T and Failure K

Suppose that the unit is replaced at time T ($0 \leq T < \infty$) or at failure K ($K = 0, 1, 2, \dots$), whichever occurs last. Then, the expected cost rate is [9]

$$C_L(T, K) = \frac{c_T + (c_K - c_T)\bar{P}_K(T) + c_M \left[H(T) + \int_T^\infty \bar{P}_K(t)h(t)dt \right]}{T + \int_T^\infty \bar{P}_K(t)dt}, \quad (28.49)$$

which agrees with (28.11) when $K = 0$ and (28.14) when $T = 0$.

When $c_T = c_K$, we find optimal T_L^* and K_L^* to minimize $C_L(T, K)$. Differentiating $C_L(T, K)$ with respect to T and setting it equal to zero,

$$h(T) \left[T + \int_T^\infty \bar{P}_K(t) dt \right] - H(T) - \int_T^\infty \bar{P}_K(t) h(t) dt = \frac{c_T}{c_M}, \tag{28.50}$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_L^* ($0 < T_L^* < \infty$) which satisfies (28.50) for any K , and the resulting cost rate is

$$C_L(T_L^*, K) = c_M h(T_L^*). \tag{28.51}$$

Furthermore, noting that the left-hand side of (28.50) increases strictly with K from that of (28.12), T_L^* decreases with K from T^* given in (28.12), and $T_L^* \leq T^*$.

Forming $C_L(T, K + 1) - C_L(T, K) \geq 0$,

$$\tilde{H}(T, K) \left[T + \int_T^\infty \bar{P}_K(t) dt \right] - H(T) - \int_T^\infty \bar{P}_K(t) h(t) dt \geq \frac{c_T}{c_M}, \tag{28.52}$$

where

$$\tilde{H}(T, K) \equiv \frac{\int_T^\infty p_K(t) h(t) dt}{\int_T^\infty p_K(t) dt},$$

which increases strictly with T from $1 / \int_0^\infty p_K(t) dt$ to $h(\infty)$ and increases strictly with K from $\tilde{H}(T, 0) = Q(T)$ in (28.20) to $h(\infty)$, and $\tilde{H}(T, K) > h(T)$ from Appendix 5. Substituting (28.50) for (28.52),

$$\tilde{H}(T, K) \geq h(T),$$

which always holds for any T , i.e., $C_L(T, K + 1) > C_L(T, K)$. Therefore, the optimal policy which minimizes $C_L(T, K)$ is $K_L^* = 0$ and $T_L^* = T^*$ given in (28.12).

Furthermore, noting that the left-hand side of (28.52) increases strictly with K from

$$TQ(T) - H(T) > Th(T) - H(T),$$

which agrees with that of (28.12), if $T \geq T^*$, then $K_L^* = 0$, and if $T < T^*$, then there exists a finite and unique minimum K_L^* ($0 \leq K_L^* < \infty$) which satisfies (28.52). In addition, because the left-hand side of (28.52) increases with T from that of (28.15), K_L^* decreases with T from K^* given in (28.15), and $K_L^* \leq K^*$.

Comparing replacement first and last with T and K , if $T < T^*$, then replacement last is better than replacement first, if $T = T^*$, then replacement with time T^* is better than replacement first and last, and if $T > T^*$, then replacement first is better than replacement last.

(3) Replacement First with Time T and Cycle N

Suppose that the unit is replaced at time T ($0 < T \leq \infty$) or at cycle N ($N = 1, 2, \dots$), whichever occurs first. Then, the expected cost rate is [7, p. 76]

$$C_F(T, N) = \frac{c_T + (c_N - c_T)G^{(N)}(T) + c_M \int_0^T [1 - G^{(N)}(t)]h(t) dt}{\int_0^T [1 - G^{(N)}(t)] dt}, \tag{28.53}$$

which agrees with (28.11) when $N = \infty$ and (28.16) when $T = \infty$.

When $c_T = c_N$ and $G(t) = 1 - e^{-\theta t}$, we find optimal T_F^* and N_F^* to minimize $C_F(T, N)$. Differentiating $C_F(T, N)$ with respect to T and setting it equal to zero,

$$h(T) \int_0^T [1 - G^{(N)}(t)] dt - \int_0^T [1 - G^{(N)}(t)]h(t) dt = \frac{c_T}{c_M}, \tag{28.54}$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_F^* ($0 < T_F^* < \infty$) which satisfies (28.54) for any N , and the resulting cost rate is

$$C_F(T_F^*, N) = c_M h(T_F^*). \tag{28.55}$$

Furthermore, noting that the left-hand side of (28.54) increases strictly with N to that of (28.12), T_F^* decreases with N to T^* given in (28.12), and $T_F^* \geq T^*$.

Forming $C_F(T, N + 1) - C_F(T, N) \geq 0$,

$$H(T, N) \int_0^T [1 - G^{(N)}(t)] dt - \int_0^T [1 - G^{(N)}(t)]h(t) dt \geq \frac{c_T}{c_M}, \tag{28.56}$$

where

$$H(T, N) \equiv \frac{\int_0^T (\theta t)^N e^{-\theta t} h(t) dt}{\int_0^T (\theta t)^N e^{-\theta t} dt},$$

which increases strictly with T to $H(N)$ and increases strictly with N to $h(T)$, and $H(T, N) < h(T)$ from Appendix 3. Substituting (28.54) for (28.56),

$$H(T, N) \geq h(T),$$

which does not hold for any T . Therefore, the optimal policy which minimizes $C_F(T, N)$ is $N_F^* = \infty$ and $T_F^* = T^*$ given in (28.12).

Furthermore, noting that the left-hand side of (28.56) increases strictly with N to that of (28.12), if $T \leq T^*$, then $N_F^* = \infty$, and if $T > T^*$, then there exists a finite and unique minimum N_F^* ($1 \leq N_F^* < \infty$) which satisfies (28.56). In addition, because the left-hand side of (28.56) increases with T to that of (28.18), N_F^* decreases with T to N^* given in (28.18), and $N_F^* \geq N^*$.

(4) Replacement Last with Time T and Cycle N

Suppose that the unit is replaced at time T ($0 \leq T < \infty$) or at cycle N ($N = 0, 1, 2, \dots$), whichever occurs last. Then, the expected cost rate is [7, p. 79]

$$C_L(T, N) = \frac{c_T + (c_N - c_T)[1 - G^{(N)}(T)] + c_M \{H(T) + \int_T^\infty [1 - G^{(N)}(t)]h(t)dt\}}{T + \int_T^\infty [1 - G^{(N)}(t)]dt}, \quad (28.57)$$

which agrees with (28.11) when $N = 0$ and (28.16) when $T = 0$,

When $c_T = c_N$ and $G(t) = 1 - e^{-\theta t}$, we find optimal T_L^* and N_L^* to minimize $C_L(T, N)$. Differentiating $C_L(T, N)$ with respect to T and setting it equal to zero,

$$h(T) \left\{ T + \int_T^\infty [1 - G^{(N)}(t)]dt \right\} - H(T) - \int_T^\infty [1 - G^{(N)}(t)]h(t)dt = \frac{c_T}{c_M}, \quad (28.58)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_L^* ($0 < T_L^* < \infty$) which satisfies (28.58) for any N , and the resulting cost rate is

$$C_L(T_L^*, N) = c_M h(T_L^*). \quad (28.59)$$

Furthermore, noting that the left-hand side of (28.58) decreases strictly with N from that of (28.12), T_L^* increases with N from T^* given in (28.12), and $T_L^* \leq T^*$.

Forming $C_L(T, N + 1) - C_L(T, N) \geq 0$,

$$\tilde{H}(T, N) \left\{ T + \int_T^\infty [1 - G^{(N)}(t)]dt \right\} - H(T) - \int_T^\infty [1 - G^{(N)}(t)]h(t)dt \geq \frac{c_T}{c_M}, \quad (28.60)$$

where

$$\tilde{H}(T, N) \equiv \frac{\int_T^\infty (\theta t)^N e^{-\theta t} h(t) dt}{\int_T^\infty (\theta t)^N e^{-\theta t} dt},$$

which increases strictly with T from $H(N)$ in (28.18) to $h(\infty)$ and increases strictly with N from $\int_0^\infty \theta e^{-\theta t} h(T + t) dt$ to $h(\infty)$, and $\tilde{H}(T, N) > h(T)$ from Appendix 3. Substituting (28.58) for (28.60),

$$\tilde{H}(T, N) \geq h(T),$$

which always holds for any T , i.e., $C_L(T, N + 1) \geq C_L(T, N)$. Therefore, the optimal policy which minimizes $C_L(T, N)$ is $N_L^* = 0$ and $T_L^* = T^*$ given in (28.12).

Furthermore, noting that the left-hand side of (28.60) increases strictly with N from

$$T \int_0^\infty \theta e^{-\theta t} h(T + t) dt - H(T) > Th(T) - H(T),$$

which agrees with that of (28.12), if $T \geq T^*$, then $N_L^* = 0$, and if $T < T^*$, then there exists a finite and unique minimum N_L^* ($0 \leq N_L^* < \infty$) which satisfies (28.60). In addition, because the left-hand side of (28.60) increases with T from that of (28.18), N_L^* decreases with T from N^* given in (28.18), and $N_L^* \leq N^*$.

Comparing replacement first and last with T and N , if $T < T^*$, then replacement last is better than replacement first, if $T = T^*$, then replacement with time T^* is better than replacement first and last, and if $T > T^*$, then replacement first is better than replacement last.

(5) Replacement Overtime First with Time T and Failure K

Suppose that the unit is replaced at failure K ($K = 1, 2, \dots$) before time T or at the first failure over time T ($0 \leq T < \infty$), whichever occurs first. Then, the expected cost rate is [8, p. 47]

$$C_{OF}(T, K) = \frac{c_{OK} + (c_K - c_{OK})P_K(T) + c_M [\bar{P}_K(T) + \int_0^T \bar{P}_K(t)h(t)dt]}{\int_0^T \bar{P}_K(t)dt + \bar{P}_K(T) \int_T^\infty e^{-H(t)+H(T)} dt}, \quad (28.61)$$

which agrees with (28.14) when $T = \infty$ and (28.19) when $K = \infty$.

When $c_K = c_{OK}$, we find optimal T_{OF}^* and K_{OF}^* to minimize $C_{OF}(T, K)$. Differentiating $C_{OF}(T, K)$ with respect to T and setting it equal to zero,

$$Q(T) \int_0^T \bar{P}_K(t)dt - \int_0^T \bar{P}_K(t)h(t)dt = \frac{c_{OK}}{c_M}, \quad (28.62)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_{OF}^* ($0 < T_{OF}^* < \infty$) which satisfies (28.62) for any K , and the resulting cost rate is

$$C_{OF}(T_{OF}^*, K) = c_M Q(T_{OF}^*) = \frac{c_{OK} + c_M \int_0^{T_{OK}^*} \bar{P}_K(t) h(t) dt}{\int_0^{T_{OF}^*} \bar{P}_K(t) dt}, \quad (28.63)$$

which agrees with (28.45) when $c_K = c_{OK} = c_T$ and $T_{OF}^* = T$. Comparing (28.62) with (28.46), $T_{OF}^* < T_F^*$ because $Q(T) > h(T)$, and from (28.63), when $c_K = c_{OK}$, replacement first is better than replacement overtime first. Furthermore, noting that the left-hand side of (28.62) increases strictly with K to that of (28.20), T_{OF}^* decreases with K to T_{OK}^* given in (28.20), and $T_{OF}^* \geq T_{OK}^*$.

Forming $C_{OF}(T, K + 1) - C_{OF}(T, K) \geq 0$,

$$H_1(T, K) \left[\int_0^T \bar{P}_K(t) dt + \bar{P}_K(T) \int_T^\infty e^{-H(t)+H(T)} dt \right] - \bar{P}_K(T) - \int_0^T \bar{P}_K(t) h(t) dt \geq \frac{c_M}{c_{OK}}, \quad (28.64)$$

where

$$H_1(T, K) \equiv \frac{P_K(T)}{\int_0^T [\int_t^\infty e^{-H(u)+H(t)} du] dP_K(t)},$$

which increases strictly with T to $1/\int_0^\infty p_K(t) dt$ and increases strictly with K to $Q(T)$, and $H_1(T, K) < Q(T)$ from Appendix 6. Substituting (28.62) for (28.64),

$$H_1(T, K) \geq Q(T),$$

which does not hold for any T , i.e., $C_{OF}(T, K + 1) < C_{OF}(T, K)$. Therefore, the optimal policy which minimizes $C_{OF}(T, K)$ is $K_{OF}^* = \infty$ and $T_{OF}^* = T_{OK}^*$ given in (28.20).

Furthermore, noting that the left-hand side of (28.64) increases with K to that of (28.20), if $T \leq T_{OK}^*$, then $K_{OF}^* = \infty$, and if $T > T_{OK}^*$, then there exists a finite and unique minimum K_{OF}^* ($1 \leq K_{OF}^* < \infty$) which satisfies (28.64). In addition, because the left-hand side of (28.64) increases with T to that of (28.15), K_{OF}^* decreases with T to K^* given in (28.15), and $K_{OF}^* \geq K^*$.

(6) Replacement Overtime Last with Time T and Failure K

Suppose that the unit is replaced at failure K ($K = 0, 1, 2, \dots$) after time T ($0 \leq T < \infty$) or at the first failure over time T , whichever occurs last. Then, the expected cost rate is [8, p. 50]

$$C_{OL}(T, K) = \frac{c_{OK} + (c_K - c_{OK}) \bar{P}_K(T) + c_M \left[H(T) + P_K(T) + \int_T^\infty \bar{P}_K(t) h(t) dt \right]}{T + \int_T^\infty \bar{P}_K(t) dt + P_K(T) \int_T^\infty e^{-H(t)+H(T)} dt}, \quad (28.65)$$

which agrees with (28.14) when $T = 0$ and (28.19) when $K = 0$, and $C_{OL}(T, 0) = C_{OL}(T, 1)$ when $c_K = c_{OK}$.

When $c_K = c_{OK}$, we find optimal T_{OL}^* and K_{OL}^* to minimize $C_{OL}(T, K)$. Differentiating $C_{OL}(T, K)$ with respect to T and setting it equal to zero,

$$Q(T) \left[T + \int_T^\infty \bar{P}_K(t) dt \right] - H(T) - \int_T^\infty \bar{P}_K(t) h(t) dt = \frac{c_{OK}}{c_M}, \quad (28.66)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique minimum T_{OL}^* ($0 < T_{OL}^* < \infty$) which satisfies (28.66) for any K , and the resulting cost rate is

$$C_{OL}(T_{OL}^*) = c_M Q(T_{OL}^*) = \frac{c_{OK} + c_M \left[H(T_{OL}^*) + \int_{T_{OL}^*}^\infty \bar{P}_K(t) h(t) dt \right]}{T_{OL}^* + \int_{T_{OL}^*}^\infty \bar{P}_K(t) dt}, \quad (28.67)$$

which agrees with (28.49) when $T_{OL}^* = T$ and $c_K = c_{OK} = c_T$. Comparing (28.66) with (28.50), $T_{OL}^* < T_L^*$ because $Q(T) > h(T)$, and from (28.67), when $c_K = c_{OK}$, replacement last is better than replacement overtime last. Furthermore, noting that the left-hand side of (28.66) increases strictly with K from (28.20), T_{OL}^* decreases with K from T_{OK}^* given in (28.20), and $T_{OL}^* \leq T_{OK}^* \leq T_{OF}^*$.

Forming $C_{OL}(T, K + 1) - C_{OL}(T, K) \geq 0$,

$$\tilde{H}_1(T, K) \left[T + \int_T^\infty \bar{P}_K(t) dt + P_K(T) \int_T^\infty e^{-H(t)+H(T)} dt \right] - H(T) - P_K(T) - \int_T^\infty \bar{P}_K(t) h(t) dt = \frac{c_{OK}}{c_M}, \quad (28.68)$$

where

$$\tilde{H}_1(T, K) \equiv \frac{\bar{P}_K(T)}{\int_T^\infty [\int_t^\infty e^{-H(u)+H(t)} du] dP_K(t)},$$

which increases strictly with T from $1/\int_0^\infty p_K(t) dt$ to $h(\infty)$ and increases strictly with K from $\tilde{H}_1(T, 1)$ to $h(\infty)$, and $\tilde{H}_1(T, K) > Q(T)$ from Appendix 6. Substituting (28.66) for (28.68),

$$\tilde{H}_1(T, K) \geq Q(T),$$

which always holds for any T , i.e., $C_{OL}(T, K + 1) \geq C_{OL}(T, K)$. Therefore, the optimal policy which minimizes $C_{OL}(T, K)$ is $K_{OL}^* = 0$ or 1 and $T_{OL}^* = T_{OK}^*$ given in (28.20).

Furthermore, noting that the left-hand side of (28.68) increases strictly with K from

$$\begin{aligned} \tilde{H}_1(T, 1) & \left[T + \int_T^\infty e^{-H(t)+H(T)} dt \right] - H(T) - 1 \\ & > TQ(T) - H(T), \end{aligned}$$

which agrees with that of (28.20), if $T \geq T_{OK}^*$, then $K_{OL}^* = 0$, and if $T < T_{OK}^*$, then there exists a finite and unique minimum K_{OL}^* ($1 \leq K_{OL}^* < \infty$) which satisfies (28.68). In addition, because the left-hand side of (28.68) increases with T from that of (28.15), K_{OL}^* decreases with T from K^* given in (28.15), and $K_{OL}^* \leq K^* \leq K_{OF}^*$.

Comparing replacement overtime first and last with T and K , if $T < T_{OK}^*$, then replacement overtime last is better than overtime first, if $T = T_{OK}^*$, then replacement overtime with T_{OK}^* is better than overtime first and last, and if $T > T_{OK}^*$, then replacement overtime first is better than overtime last.

(7) Replacement Overtime First with Time T and Cycle N

Suppose that the unit is replaced at cycle N ($N = 1, 2, \dots$) before time T or at the first cycle over time T ($0 \leq T \leq \infty$), whichever occurs first. Then, the expected cost rate is [8, p. 42]

$$\begin{aligned} C_{OF}(T, N) = & \frac{c_{ON} + (c_N - c_{ON})G^{(N)}(T) + c_M \left\{ \int_0^T [1 - G^{(N)}(t)]h(t)dt + \sum_{j=0}^{N-1} \int_0^T \int_{T-t}^T \bar{G}(u)h(t+u)du \right\} dG^{(j)}(t)}{\int_0^T [1 - G^{(N)}(t)]dt + \sum_{j=0}^{N-1} \int_0^T \int_{T-t}^\infty \bar{G}(u)du dG^{(j)}(t)}, \end{aligned} \quad (28.69)$$

which agrees with (28.16) when $T = \infty$ and (28.22) when $N = \infty$.

When $c_N = c_{ON}$ and $G(t) = 1 - e^{-\theta t}$, (28.69) is

$$\begin{aligned} C_{OF}(T, N) = & \frac{c_{ON} + c_M \left\{ \int_0^T [1 - G^{(N)}(t)]h(t)dt + [1 - G^{(N)}(T)] \int_0^\infty e^{-\theta t} h(T+t)dt \right\}}{\int_0^T [1 - G^{(N)}(t)]dt + (1/\theta)[1 - G^{(N)}(T)]}. \end{aligned} \quad (28.70)$$

We find optimal T_{OF}^* and N_{OF}^* to minimize $C_{OF}(T, N)$. Differentiating $C_{OF}(T, N)$ with respect to T and setting it equal to zero,

$$\begin{aligned} & \int_0^\infty \theta e^{-\theta t} h(T+t)dt \int_0^T [1 - G^{(N)}(t)]dt \\ & - \int_0^T [1 - G^{(N)}(t)]h(t)dt = \frac{c_{ON}}{c_M}, \end{aligned} \quad (28.71)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_{OF}^* ($0 < T_{OF}^* < \infty$) which satisfies (28.71) for any N , and the resulting cost rate is

$$\begin{aligned} C_{OF}(T_{OF}^*, N) & = c_M \int_0^\infty \theta e^{-\theta t} h(T_{OF}^* + t)dt \\ & = \frac{c_{ON} + c_M \int_0^{T_{OF}^*} [1 - G^{(N)}(t)]h(t)dt}{\int_0^{T_{OF}^*} [1 - G^{(N)}(t)]dt}, \end{aligned} \quad (28.72)$$

which agrees with (28.53) when $c_N = c_{ON} = c_T$ and $T_{OF}^* = T$. Comparing (28.71) with (28.54), $T_{OF}^* < T_F^*$ because $\int_0^\infty \theta e^{-\theta t} h(T+t)dt > h(T)$, and from (28.72), when $c_N = c_{ON}$, replacement first is better than replacement overtime first. Furthermore, noting that the left-hand side of (28.71) increases strictly with N to that of (28.24), T_{OF}^* decreases with N to T_{ON}^* given in (28.24), and $T_{OF}^* \geq T_{ON}^*$.

Forming $C_{OF}(T, N + 1) - C_{OF}(T, N) \geq 0$,

$$\begin{aligned} & \frac{H_1(T, N)}{\theta} \sum_{j=0}^{N-1} G^{(j)}(T) - \int_0^T [1 - G^{(N)}(t)]h(t)dt \\ & - [1 - G^{(N)}(T)] \int_0^\infty e^{-\theta t} h(T+t)dt \geq \frac{c_{ON}}{c_M}, \end{aligned} \quad (28.73)$$

where

$$H_1(T, N) \equiv \frac{\int_0^T \int_0^\infty \theta e^{-\theta u} h(t+u)du dG^{(N)}(t)}{G^{(N)}(T)},$$

which increases strictly with T to $H(N)$ and increases strictly with N to $\int_0^\infty \theta e^{-\theta t} h(T+t)dt$, and $H_1(T, N) < \int_0^\infty \theta e^{-\theta t} h(T+t)dt$ from Appendix 7. Substituting (28.71) for (28.73),

$$H_1(T, N) \geq \int_0^\infty \theta e^{-\theta t} h(T+t)dt,$$

which does not hold for any T . Therefore, the optimal policy which minimizes $C_{OF}(T, N)$ is $N_{OF}^* = \infty$ and $T_{OF}^* = T_{ON}^*$ given in (28.24).

Furthermore, noting that the left-hand side of (28.73) increases strictly with N to that of (28.24), if $T \leq T_{ON}^*$, then $N_{OF}^* = \infty$, and if $T > T_{ON}^*$, then there exists a finite and unique minimum N_{OF}^* ($1 \leq N_{OF}^* < \infty$) which satisfies (28.73) and approaches to N^* given in (28.5) as $T \rightarrow \infty$.

(8) Replacement Overtime Last with Time T and Cycle N

Suppose that the unit is replaced at cycle N ($N = 0, 1, 2, \dots$) or at the first cycle over time T ($0 \leq T < \infty$), whichever occurs last. Then, the expected cost rate is [8, p. 44]

$$C_{OL}(T, N) = \frac{c_{ON} + (c_N - c_{ON})[1 - G^{(N)}(T)] + c_M \left\{ \int_0^\infty [1 - G^{(N)}(t)]h(t)dt + \sum_{j=N}^\infty \int_0^T \int_0^\infty \bar{G}(u)h(t+u)du \right\} dG^{(j)}(t)}{(1/\theta)[N + \sum_{j=N}^\infty G^{(j)}(T)]}, \quad (28.74)$$

which agrees with (28.16) when $T = 0$ and (28.22) when $N = 0$, and $C_{OL}(T, 0) = C_{OL}(T, 1)$ when $c_N = c_{ON}$.

When $c_N = c_{ON}$ and $G(t) = 1 - e^{-\theta t}$, (28.74) is

$$C_{OL}(T, N) = \frac{c_{ON} + c_M \left\{ H(T) + \int_T^\infty [1 - G^{(N)}(t)]h(t)dt + G^{(N)}(T) \int_0^\infty e^{-\theta t} h(T+t)dt \right\}}{(1/\theta)[N + \sum_{j=N}^\infty G^{(j)}(T)]}. \quad (28.75)$$

We find optimal T_{OL}^* and N_{OL}^* to minimize $C_{OL}(T, N)$. Differentiating $C_{OL}(T, N)$ with respect to T and setting it equal to zero,

$$\int_0^\infty e^{-\theta t} h(T+t)dt \left[N + \sum_{j=N+1}^\infty G^{(j)}(T) \right] - H(T) - \int_T^\infty [1 - G^{(N)}(t)]h(t)dt = \frac{c_{ON}}{c_M}, \quad (28.76)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_{OL}^* ($0 < T_{OL}^* < \infty$) which satisfies (28.76) for any N , and the resulting cost rate is

$$C_{OL}(T_{OL}^*, N) = c_M \int_0^\infty \theta e^{-\theta t} h(T_{OL}^* + t)dt = \frac{c_{ON} + c_M \left\{ H(T_{OL}^*) + \int_{T_{OL}^*}^\infty [1 - G^{(N)}(t)]h(t)dt \right\}}{(1/\theta)[N + \sum_{j=N+1}^\infty G^{(j)}(T_{OL}^*)]}, \quad (28.77)$$

which agrees with (28.57) when $T_{OL}^* = T$ and $c_N = c_{ON} = c_T$. Comparing (28.76) with (28.58), $T_{OL}^* < T_L^*$ because $\int_0^\infty \theta e^{-\theta t} h(T+t)dt > h(T)$, and from (28.77), when $c_N = c_{ON}$, replacement last is better than replacement overtime last. Furthermore, noting that the left-hand side of (28.76) increases strictly with N from (28.24), T_{OL}^* decreases with N from T_{ON}^* given in (28.24), and $T_{OL}^* \leq T_{ON}^* \leq T_{OF}^*$.

Forming $C_{OL}(T, N+1) - C_{OL}(T, N) \geq 0$,

$$\frac{\tilde{H}_1(T, N)}{\theta} \left[N + \sum_{j=N+1}^\infty G^{(j)}(T) \right] - H(T) - \int_T^\infty [1 - G^{(N)}(t)]h(t)dt \geq \frac{c_{ON}}{c_M}, \quad (28.78)$$

where

$$\tilde{H}_1(T, N) \equiv \frac{\int_T^\infty \left[\int_0^\infty \theta e^{-\theta u} h(t+u)du \right] dG^{(N)}(t)}{1 - G^{(N)}(T)},$$

which increases strictly with T from $H(N)$ to $h(\infty)$ and increases strictly with N from $\tilde{H}_1(T, 1)$ to $h(\infty)$, and $\tilde{H}_1(T, N) > \int_0^\infty \theta e^{-\theta t} h(T+t)dt$ from Appendix 7. Substituting (28.76) for (28.78),

$$\tilde{H}_1(T, N) \geq \int_0^\infty \theta e^{-\theta t} h(T+t)dt,$$

which always holds for any T . Therefore, the optimal policy which minimizes $C_{OL}(T, N)$ is $N_{OL}^* = 0$ or 1 and $T_{OL}^* = T_{ON}^*$ given in (28.24).

Furthermore, noting that the left-hand side of (28.78) increases strictly with N from

$$\frac{\tilde{H}_1(T, 1)}{\theta} [\theta T + e^{-\theta T}] - H(T) - \int_T^\infty e^{-\theta t} h(t)dt > T \int_0^\infty \theta e^{-\theta t} h(T+t)dt - H(T),$$

which agrees with (28.24), if $T \geq T_{ON}^*$, then $N_{OL}^* = 0$, and if $T < T_{ON}^*$, then there exists a finite and unique minimum N_{OL}^* ($1 \leq N_{OL}^* < \infty$) which satisfies (28.78). In addition, because the left-hand side of (28.78) increases strictly with T from

$$N \int_0^\infty \frac{\theta(\theta t)^N}{N!} e^{-\theta t} h(t)dt - \int_0^\infty [1 - G^{(N)}(t)]h(t)dt,$$

which agrees with that of (28.18), N_{OL}^* decreases with T from N^* , and $N_{OL}^* \leq N^*$.

Comparing replacement overtime first and last with T and N , if $T < T_{ON}^*$, then replacement overtime last is better than overtime first, if $T = T_{ON}^*$, then replacement overtime with T_{ON}^* is better than replacement overtime first and last, and if $T > T_{ON}^*$, then replacement overtime first is better than overtime last.

(9) Replacement First with Failure K and Cycle N

Suppose that the unit is replaced at failure K ($K = 1, 2, \dots$) or at cycle N ($N = 1, 2, \dots$), whichever occurs first. Then, the expected cost rate is [10]

$$C_F(K, N) = \frac{c_K - (c_K - c_N) \int_0^\infty \bar{P}_K(t) dG^{(N)}(t) + c_M \int_0^\infty [1 - G^{(N)}(t)] \bar{P}_K(t) h(t) dt}{\int_0^\infty [1 - G^{(N)}(t)] \bar{P}_K(t) dt}, \quad (28.79)$$

which agrees with (28.14) when $N = \infty$ and (28.16) when $K = \infty$.

We find optimal K_F^* and N_F^* to minimize $C_F(K, N)$ when $c_K = c_N$ and $G(t) = 1 - e^{-\theta t}$. Then, (28.79) is

$$C_F(K, N) = \frac{c_K + c_M \int_0^\infty [1 - G^{(N)}(t)] \bar{P}_K(t) h(t) dt}{\int_0^\infty [1 - G^{(N)}(t)] \bar{P}_K(t) dt}. \quad (28.80)$$

Forming $C_F(K + 1, N) - C_F(K, N) \geq 0$,

$$H_1(K, N) \int_0^\infty [1 - G^{(N)}(t)] \bar{P}_K(t) dt - \int_0^\infty [1 - G^{(N)}(t)] \bar{P}_K(t) h(t) dt \geq \frac{c_K}{c_M}, \quad (28.81)$$

where

$$H_1(K, N) \equiv \frac{\int_0^\infty [1 - G^{(N)}(t)] p_K(t) h(t) dt}{\int_0^\infty [1 - G^{(N)}(t)] p_K(t) dt},$$

which increases strictly with K to $h(\infty)$ and increases strictly with N to $1 / \int_0^\infty p_K(t) dt$ from Appendix 8. Thus, because the left-hand side of (28.81) increases strictly with K to ∞ , there exists a finite and unique minimum K_F^* ($1 \leq K_F^* < \infty$) which satisfies (28.81) for any N . In addition, noting that the left-hand side of (28.81) goes to that of (28.15) as $N \rightarrow \infty$, K_F^* approaches to K^* given in (28.15) as $N \rightarrow \infty$.

Forming $C_F(K, N + 1) - C_F(K, N) \geq 0$,

$$H_2(K, N) \int_0^\infty [1 - G^{(N)}(t)] \bar{P}_K(t) dt - \int_0^\infty [1 - G^{(N)}(t)] \bar{P}_K(t) h(t) dt \geq \frac{c_N}{c_M}, \quad (28.82)$$

where

$$H_2(K, N) \equiv \frac{\int_0^\infty (\theta t)^N e^{-\theta t} \bar{P}_K(t) h(t) dt}{\int_0^\infty (\theta t)^N e^{-\theta t} \bar{P}_K(t) dt},$$

which increases strictly with N to $h(\infty)$ and increases strictly with K to $H(N)$ from Appendix 9. Thus, because the left-hand side of (28.82) increases strictly with N to ∞ , there exists a finite and unique minimum N_F^* ($1 \leq N_F^* < \infty$) which satisfies (28.82) for any K . In addition, noting that the

left-hand side of (28.82) goes to that of (28.18) as $K \rightarrow \infty$, N_F^* approaches to N^* given in (28.18) as $K \rightarrow \infty$.

(10) Replacement Last with Failure K and Cycle N

Suppose that the unit is replaced at failure K ($K = 0, 1, 2, \dots$) or at cycle N ($N = 0, 1, 2, \dots$), whichever occurs last. Then, the expected cost rate is [10]

$$C_L(K, N) = \frac{c_K - (c_K - c_N) \int_0^\infty P_K(t) dG^{(N)}(t) + c_M \int_0^\infty [1 - G^{(N)}(t)] P_K(t) h(t) dt}{\int_0^\infty [1 - G^{(N)}(t)] P_K(t) dt}, \quad (28.83)$$

which agrees with (28.14) when $N = 0$ and (28.16) when $K = 0$.

We find optimal K_L^* and N_L^* to minimize $C_L(K, N)$ when $c_K = c_N$ and $G(t) = 1 - e^{-\theta t}$. Then, (28.83) is

$$C_L(K, N) = \frac{c_K + c_M \int_0^\infty [1 - G^{(N)}(t)] P_K(t) h(t) dt}{\int_0^\infty [1 - G^{(N)}(t)] P_K(t) dt}. \quad (28.84)$$

Forming $C_L(K + 1, N) - C_L(K, N) \geq 0$,

$$H_3(K, N) \int_0^\infty [1 - G^{(N)}(t)] P_K(t) dt - \int_0^\infty [1 - G^{(N)}(t)] P_K(t) h(t) dt \geq \frac{c_K}{c_M}, \quad (28.85)$$

where

$$H_3(K, N) \equiv \frac{\int_0^\infty G^{(N)}(t) p_K(t) h(t) dt}{\int_0^\infty G^{(N)}(t) p_K(t) dt},$$

which increases strictly with K to $h(\infty)$ and increases strictly with N from $1 / \int_0^\infty p_K(t) dt$ to $h(\infty)$ from Appendix 9. Thus, because the left-hand side of (28.85) increases strictly with K to ∞ , there exists a finite and unique K_L^* ($0 \leq K_L^* < \infty$) which satisfies (28.85) for any N . In addition, noting that the left-hand side of (28.85) agrees with that of (28.15) when $N = 0$, $K_L^* = K^*$ given in (28.15) when $N = 0$.

Forming $C_L(K, N + 1) - C_L(K, N) \geq 0$,

$$H_4(K, N) \int_0^\infty [1 - G^{(N)}(t)] P_K(t) dt - \int_0^\infty [1 - G^{(N)}(t)] P_K(t) h(t) dt \geq \frac{c_K}{c_M}, \quad (28.86)$$

where

$$H_4(K, N) \equiv \frac{\int_0^\infty (\theta t)^N e^{-\theta t} P_K(t) h(t) dt}{\int_0^\infty (\theta t)^N e^{-\theta t} P_K(t) dt},$$

which increases with N to $h(\infty)$ and increases strictly with K from $H(N)$ to $h(\infty)$ from Appendix 9. Thus, because the left-hand side of (28.86) increases strictly with N to ∞ , there exists a finite and unique minimum N_L^* ($0 \leq N_L^* < \infty$) which satisfies (28.86) for any K . In addition, noting that the left-hand side of (28.86) agrees with that of (28.18) when $K = 0$, $N_L^* = N^*$ given in (28.18) when $K = 0$.

As modified policies of periodic replacements with two variables, we consider the following two policies:

(11) Modified Replacement with Time T after Failure K

Suppose that the unit is replaced at time T ($0 \leq T < \infty$) after K th ($K = 0, 1, 2, \dots$) failure. Then, the mean time to replacement is

$$\int_0^\infty (t + T)dP_K(t) = T + \int_0^\infty \bar{P}_K(t)dt,$$

and the expected number of failures until replacement is

$$\begin{aligned} & \int_0^\infty [K + H(T + t) - H(t)]dP_K(t) \\ &= \int_0^\infty H(T + t)dP_K(t) = \int_0^\infty h(T + t)\bar{P}_K(t)dt. \end{aligned}$$

Thus, the expected cost rate is

$$C_{M1}(T, K) = \frac{c_T + c_M \int_0^\infty H(T + t)dP_K(t)}{T + \int_0^\infty \bar{P}_K(t)dt}, \quad (28.87)$$

which agrees with (28.11) when $K = 0$ and (28.14) when $T = 0$.

We find optimal T_{M1}^* and K_{M1}^* to minimize $C_{M1}(T, K)$. Differentiating $C_{M1}(T, K)$ with respect to T and setting it equal to zero,

$$\begin{aligned} & \int_0^\infty h(T + t)dP_K(t) \left[T + \int_0^\infty \bar{P}_K(t)dt \right] \\ & - \int_0^\infty H(T + t)dP_K(t) = \frac{c_T}{c_M}, \end{aligned} \quad (28.88)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_{M1}^* ($0 \leq T_{M1}^* < \infty$) which satisfies (28.88) for any K , and the resulting cost rate is

$$C_{M1}(T_{M1}^*) = c_M \int_0^\infty h(T_{M1}^* + t)dP_K(t). \quad (28.89)$$

In addition, noting that the left-hand side of (28.88) increases strictly with K from that of (28.12) to ∞ , T_{M1}^* decreases with K from T^* given in (28.12), and $T_{M1}^* \leq T^*$.

Forming $C_{M1}(T, K + 1) - C_{M1}(T, K) \geq 0$,

$$\begin{aligned} & H_{M1}(T, K) \left[T + \int_0^\infty \bar{P}_K(t)dt \right] \\ & - \int_0^\infty H(T + t)dP_K(t) \geq \frac{c_T}{c_M}, \end{aligned} \quad (28.90)$$

where

$$H_{M1}(T, K) \equiv \frac{\int_0^\infty h(T + t)p_K(t)dt}{\int_0^\infty p_K(t)dt},$$

which increases strictly with T from $1/\int_0^\infty p_K(t)dt$ to $h(\infty)$ and increases strictly with K to $h(\infty)$ from Appendix 10. Thus, because the left-hand side of (28.90) increases strictly with K to ∞ , there exists a finite and unique minimum K_{M1}^* ($0 \leq K_{M1}^* < \infty$) which satisfies (28.90) for any T . Noting that the left-hand side of (28.90) increases strictly with T from that of (28.15), K_{M1}^* decreases with T from K^* given in (28.15), and $K_{M1}^* \leq K^*$.

(12) Modified Replacement with Failure K after Time T

Suppose that the unit is replaced at failure K ($K = 0, 1, 2, \dots$) after time T ($0 \leq T < \infty$). Then, the mean time to replacement is

$$T + \int_T^\infty (t - T)dP_K(t, T) = T + \int_T^\infty \bar{P}_K(t, T)dt,$$

where for $T \leq t < \infty$,

$$p_K(t, T) \equiv \frac{[H(t) - H(T)]^K}{K!} e^{-[H(t) - H(T)]},$$

$$P_K(t, T) \equiv \sum_{j=K}^\infty p_j(t, T) \quad (K = 0, 1, 2, \dots),$$

and the expected number of failures until replacement is

$$H(T) + K.$$

Thus, the expected cost rate is

$$C_{M2}(T, K) = \frac{c_K + c_M[H(T) + K]}{T + \int_T^\infty \bar{P}_K(t, T)dt}, \quad (28.91)$$

which agrees with (28.11) when $K = 0$, (28.14) when $T = 0$, and (28.19) when $K = 1$.

We find optimal T_{M2}^* and K_{M2}^* to minimize $C_{M2}(T, K)$. Differentiating $C_{M2}(T, K)$ with respect to T and setting it equal to zero,

$$H_{M2}(T, K - 1) \left[T + \int_T^\infty \bar{P}_K(t, T) dt \right] - H(T) - K = \frac{c_K}{c_M}, \tag{28.92}$$

where

$$H_{M2}(T, K) \equiv \begin{cases} \frac{1}{\int_T^\infty p_K(t, T) dt} & (K = 0, 1, 2, \dots), \\ h(T) & (K = -1), \end{cases}$$

which increases strictly with K from $h(T)$ to $h(\infty)$ and increases strictly with T from $1/\int_0^\infty p_K(t) dt$ to $h(\infty)$ from Appendix 11. Thus, because the left-hand side of (28.92) increases strictly with T to ∞ , there exists a finite and unique T_{M2}^* ($0 \leq T_{M2}^* < \infty$) which satisfies (28.92) for any K , and the resulting cost rate is

$$C_{M2}(T_{M2}^*, K) = c_M H_{M2}(T_{M2}^*, K - 1). \tag{28.93}$$

In addition, noting that the left-hand side of (28.92) increases strictly with K from that of (28.12), T_{M2}^* decreases with K from T^* given in (28.12), and $T_{M2}^* \leq T^*$.

Forming $C_{M2}(T, K + 1) - C_{M2}(T, K) \geq 0$,

$$H_{M2}(T, K) \left[T + \int_T^\infty \bar{P}_K(t, T) dt \right] - H(T) - K \geq \frac{c_K}{c_M}. \tag{28.94}$$

Substituting (28.92) for (28.94),

$$H_{M2}(T, K) \geq H_{M2}(T, K - 1),$$

which always holds for any T . Thus, the optimal policy which minimizes $C_{M2}(T, K)$ is $K_{M2}^* = 0$ and $T_{M2}^* = T^*$ given in (28.12).

Therefore, because the left-hand side of (28.94) increases strictly with K from that of (28.12), if $T \geq T^*$, then $K_{M2}^* = 0$, and if $T < T^*$, then there exists a finite and unique minimum K_{M2}^* ($1 \leq K_{M2}^* < \infty$) which satisfies (28.94). In addition, noting that the left-hand side of (28.94) increases with T from that of (28.15), K_{M2}^* decreases with T from K^* given in (28.15), and $K_{M2}^* \leq K^*$.

28.4 Three Policies

Combining periodic replacement with time T , failure K , and cycle N , we consider five replacement policies. The expected cost rates are obtained, and when the replacement costs c_T , c_N , and c_K are equal, optimal policies are discussed. Using these results, we could derive optimal policies for different replacement costs.

28.4.1 Replacement First and Last

Suppose that the unit is replaced at time T ($0 < T \leq \infty$), at failure K ($K = 1, 2, \dots$) or at cycle N ($N = 1, 2, \dots$), whichever occurs first. Then, the expected cost rate is [8, p. 87]

$$C_F(T, K, N) = \frac{c_T + (c_N - c_T) \int_0^T \bar{P}_K(t) dG^{(N)}(t) + (c_K - c_T) \int_0^T [1 - G^{(N)}(t)] dP_K(t) + c_M \int_0^T [1 - G^{(N)}(t)] \bar{P}_K(t) h(t) dt}{\int_0^T [1 - G^{(N)}(t)] \bar{P}_K(t) dt}. \tag{28.95}$$

Clearly, $C_F(T, \infty, \infty) = C(T)$ in (28.11), $C_F(\infty, K, \infty) = C(K)$ in (28.14), $C_F(\infty, \infty, N) = C(N)$ in (28.16), $C_F(T, K, \infty) = C_F(T, K)$ in (28.45), $C_F(T, \infty, N) = C_F(T, N)$ in (28.53), and $C_F(\infty, K, N) = C_F(K, N)$ in (28.79).

Differentiating $C_F(T, K, N)$ with respect to T when $c_T = c_K = c_N$ and setting it equal to zero,

$$h(T) \int_0^T [1 - G^{(N)}(t)] \bar{P}_K(t) dt - \int_0^T [1 - G^{(N)}(t)] \bar{P}_K(t) h(t) dt = \frac{c_T}{c_M}, \tag{28.96}$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_F^* ($0 < T_F^* < \infty$) which satisfies (28.96) for any K and N , and the resulting cost rate is

$$C_F(T_F^*, K, N) = c_M h(T_F^*). \tag{28.97}$$

Noting that the left-hand side of (28.96) increases with K and N , T_F^* decreases with K and N , and T_F^* goes to T^* as $K \rightarrow \infty$ and $N \rightarrow \infty$.

Furthermore, by making similar discussions in Sect. 28.3, the optimal policy to minimize $C_F(T, K, N)$ is $K_F^* = N_F^* = \infty$ and $T_F^* = T^*$ given in (28.12).

Next, suppose that the unit is replaced at time T ($0 \leq T < \infty$), at failure K ($K = 0, 1, 2, \dots$) or at cycle N ($N = 0, 1, 2, \dots$), whichever occurs last. Then, the expected cost rate is [8, p. 88]

$$C_L(T, K, N) = \frac{c_T + (c_N - c_T) \int_T^\infty P_K(t) dG^{(N)}(t) + (c_K - c_T) \int_T^\infty G^{(N)}(t) dP_K(t) + c_M \{ H(T) + \int_T^\infty [1 - G^{(N)}(t)] P_K(t) h(t) dt \}}{T + \int_T^\infty [1 - G^{(N)}(t)] P_K(t) dt}. \tag{28.98}$$

Clearly, $C_L(T, 0, 0) = C(T)$ in (28.11), $C_L(0, K, 0) = C(K)$ in (28.14), $C_L(0, 0, N) = C(N)$ in (28.16), $C_L(T, K, 0) = C_L(T, K)$ in (28.49), $C_L(T, 0, N) = C_L(T, N)$ in (28.57), and $C_L(0, K, N) = C_L(K, N)$ in (28.83).

Differentiating $C_L(T, K, N)$ with respect to T when $c_T = c_K = c_N$ and setting it equal to zero,

$$h(T) \left\{ T + \int_T^\infty [1 - G^{(N)}(t)P_K(t)]dt \right\} - H(T) - \int_T^\infty [1 - G^{(N)}(t)P_K(t)]h(t)dt = \frac{c_T}{c_M}, \quad (28.99)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_L^* ($0 < T_L^* < \infty$) which satisfies (28.99) for any K and N , and the resulting cost rate is

$$C_L(T_L^*) = c_M h(T_L^*). \quad (28.100)$$

Noting that the left-hand side of (28.99) decreases with K and N , T_L^* increases with K and N , and $T_L^* = T^*$ given in (28.12) when $K = N = 0$.

Furthermore, by making similar discussions in Sect. 28.3, the optimal policy to minimize $C_L(T, K, N)$ is $K_L^* = N_L^* = 0$ and $T_L^* = T^*$ given in (28.12).

28.4.2 Replacement Middle

Suppose that the unit is replaced at time T ($0 \leq T < \infty$), at cycle K ($K = 0, 1, 2, \dots$) or at cycle N ($N = 0, 1, 2, \dots$), whichever occurs middle. That is, when t_K and t_N are denoted by the respective occurrence times of failure K and cycle N , the unit is replaced at time T for $\{t_N < T < t_K\}$ or $\{t_K < T < t_N\}$, at failure K for $\{T < t_K < N\}$ or $\{N < t_K < T\}$, and at cycle N for $\{T < t_N < t_K\}$ or $\{t_K < t_N < T\}$. Then, the expected cost rate is [5, 8, p. 90]

$$C_M(T, K, N) = \frac{c_T + (c_N - c_T) \left[\int_T^\infty \bar{P}_K(t) dG^{(N)}(t) + \int_0^T P_K(t) dG^{(N)}(t) \right] + (c_K - c_T) \left\{ \int_T^\infty [1 - G^{(N)}(t)] dP_K(t) + \int_0^T G^{(N)}(t) dP_K(t) \right\} + c_M \left\{ \int_0^T [1 - G^{(N)}(t)P_K(t)]h(t)dt + \int_T^\infty [1 - G^{(N)}(t)]\bar{P}_K(t)h(t)dt \right\}}{\int_0^T [1 - G^{(N)}(t)P_K(t)]dt + \int_T^\infty [1 - G^{(N)}(t)]\bar{P}_K(t)dt}. \quad (28.101)$$

Clearly,

$$\begin{aligned} C_M(0, K, N) &= C_F(\infty, K, N), \\ C_M(\infty, K, N) &= C_L(0, K, N), \\ C_M(T, 0, N) &= C_F(T, \infty, N), \\ C_M(T, \infty, N) &= C_L(T, 0, N), \\ C_M(T, K, 0) &= C_F(T, K, \infty), \\ C_M(T, K, \infty) &= C_L(T, K, 0). \end{aligned}$$

Differentiating $C_M(T, K, N)$ with respect to T when $c_T = c_K = c_N$ and setting it equal to zero,

$$h(T) \left\{ \int_0^T [1 - G^{(N)}(t)P_K(t)]dt + \int_T^\infty [1 - G^{(N)}(t)]\bar{P}_K(t)dt \right\} - \int_0^T [1 - G^{(N)}(t)P_K(t)]h(t)dt - \int_T^\infty [1 - G^{(N)}(t)]\bar{P}_K(t)h(t)dt = \frac{c_M}{c_T}, \quad (28.102)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_M^* ($0 < T_M^* < \infty$) which satisfies (28.102) for any K and N , and the resulting cost rate is

$$C_M(T_M^*, K, N) = c_M h(T_M^*). \quad (28.103)$$

28.4.3 Modified Replacement First and Last

Suppose that the unit is replaced at time T ($0 < T \leq \infty$) or at $\text{Max}\{t_K, t_N\}$ ($K, N = 1, 2, \dots$), whichever occurs first. Then, the probability that the unit is replaced at time T is

$$1 - G^{(N)}(T)P_K(T),$$

the probability that it is replaced at failure K is

$$\int_0^T G^{(N)}(t) dP_K(t),$$

and the probability that it is replaced at cycle N is

$$\int_0^T P_K(t) dG^{(N)}(t).$$

Thus, the mean time to replacement is

$$\begin{aligned} T[1 - G^{(N)}(T)P_K(T)] + \int_0^T tP_K(t) dG^{(N)}(t) + \int_0^T tG^{(N)}(t) dP_K(t) \\ = \int_0^T [1 - G^{(N)}(t)P_K(t)]dt, \end{aligned}$$

and the expected number of failures until replacement is

$$\begin{aligned} H(T)[1 - G^{(N)}(T)P_K(T)] + \int_0^T H(t)P_K(t) dG^{(N)}(t) + \int_0^T H(t)G^{(N)}(t) dP_K(t) \\ = \int_0^T [1 - G^{(N)}(t)P_K(t)]h(t)dt. \end{aligned}$$

Thus, the expected cost rate is

$$C_{MF}(T, K, N) = \frac{c_T + (c_K - c_T) \int_0^T G^{(N)}(t) dP_K(t) + (c_N - c_T) \int_0^T P_K(t) dG^{(N)}(t) + c_M \int_0^T [1 - G^{(N)}(t) P_K(t)] h(t) dt}{\int_0^T [1 - G^{(N)}(t) P_K(t)] dt}. \quad (28.104)$$

Clearly,

$$\begin{aligned} C_{MF}(\infty, K, N) &= C_L(0, K, N) = C_M(\infty, K, N), \\ C_{MF}(T, 0, N) &= C_F(T, \infty, N) = C_M(T, 0, N), \\ C_{MF}(T, K, 0) &= C_F(T, K, \infty) = C_M(T, K, 0). \end{aligned}$$

Differentiating $C_{MF}(T, K, N)$ with respect to T when $c_T = c_K = c_N$ and setting it equal to zero,

$$h(T) \int_0^T [1 - G^{(N)}(t) P_K(t)] dt - \int_0^T [1 - G^{(N)}(t) P_K(t)] h(t) dt = \frac{c_T}{c_M}, \quad (28.105)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_{MF}^* ($0 < T_{MF}^* < \infty$) which satisfies (28.105) for any K and N , and the resulting cost rate is

$$C_{MF}(T_{MF}^*, K, N) = c_M h(T_{MF}^*). \quad (28.106)$$

Next, suppose that the unit is replaced at time T ($0 \leq T < \infty$) or at $\text{Min}\{t_K, t_N\}$ ($K, N = 0, 1, 2, \dots$), whichever occurs last. Then, the probability that the unit is replaced at time T is

$$P_K(T) + G^{(N)}(T) \bar{P}_K(T),$$

the probability that it is replaced at failure K is

$$\int_T^\infty [1 - G^{(N)}(t)] dP_K(t),$$

and the probability that it is replaced at cycle N is

$$\int_T^\infty \bar{P}_K(t) dG^{(N)}(t).$$

Thus, the mean time to replacement is

$$T[P_K(T) + G^{(N)}(T) \bar{P}_K(T)] + \int_T^\infty t[1 - G^{(N)}(t)] dP_K(t)$$

$$\begin{aligned} &+ \int_T^\infty t \bar{P}_K(t) dG^{(N)}(t) \\ &= T + \int_T^\infty \{1 - [1 - G^{(N)}(t)] \bar{P}_K(t)\} dt, \end{aligned}$$

and the expected number of failures until replacement is

$$\begin{aligned} &H(T)[P_K(T) + G^{(N)}(T) \bar{P}_K(T)] \\ &+ \int_T^\infty H(t)[1 - G^{(N)}(t)] dP_K(t) \\ &+ \int_T^\infty H(t) \bar{P}_K(t) dG^{(N)}(t) \\ &= H(T) + \int_T^\infty \{1 - [1 - G^{(N)}(t)] \bar{P}_K(t)\} h(t) dt. \end{aligned}$$

Thus, the expected cost rate is

$$C_{ML}(T, K, N) = \frac{c_T + (c_K - c_T) \int_T^\infty [1 - G^{(N)}(t)] dP_K(t) + (c_N - c_T) \int_T^\infty \bar{P}_K(t) dG^{(N)}(t) + c_M \left(H(T) + \int_T^\infty \{1 - [1 - G^{(N)}(t)] \bar{P}_K(t)\} h(t) dt \right)}{T + \int_T^\infty \{1 - [1 - G^{(N)}(t)] \bar{P}_K(t)\} dt}. \quad (28.107)$$

Clearly,

$$\begin{aligned} C_{ML}(0, K, N) &= C_L(0, K, N) = C_M(\infty, K, N), \\ C_{ML}(T, \infty, N) &= C_F(T, \infty, N) = C_M(T, 0, N), \\ C_{ML}(T, K, \infty) &= C_F(T, K, \infty) = C_M(T, K, 0). \end{aligned}$$

Differentiating $C_{ML}(T, K, N)$ with respect to T when $c_T = c_K = c_N$ and setting it equal to zero,

$$\begin{aligned} &h(T) \left(T + \int_T^\infty \{1 - [1 - G^{(N)}(t)] \bar{P}_K(t)\} dt \right) \\ &- H(T) - \int_T^\infty \{1 - [1 - G^{(N)}(t)] \bar{P}_K(t)\} h(t) dt \\ &= \frac{c_T}{c_M}, \end{aligned} \quad (28.108)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_{ML}^* ($0 < T_{ML}^* < \infty$) which satisfies (28.108) for any K and N , and the resulting cost rate is

$$C_{ML}(T_{ML}^*) = c_M h(T_{ML}^*). \quad (28.109)$$

28.5 Redundant Replacement Policies

We have introduced the age and periodic replacements with one, two, and three policies. Using these results, we can extend them to the following replacements with time T and n number of policies, which is called *redundant replacement*

policy: Assume that the unit is replaced at time T ($0 \leq T \leq \infty$) or random times Y_1, Y_2, \dots, Y_n ($n = 1, 2, \dots$), where Y_j is independent with each other and has a general distribution $G_j(t) \equiv \Pr\{Y_j \leq t\}$ with finite mean $1/\theta_j$ ($0 < \theta_j < \infty$) [6, 8, p. 100].

28.5.1 Age Replacement

(1) Replacement First

Suppose that the unit is replaced preventively at time T ($0 < T \leq \infty$) or at time Y_1, Y_2, \dots, Y_n ($n = 1, 2, \dots$), whichever occurs first. Then, by setting $Z_m = \text{Min}\{T, Y_1, \dots, Y_n\}$, Z_m has a distribution

$$G_F(t) \equiv \Pr\{Z_m \leq t\} = \begin{cases} 1 - \prod_{j=1}^n \bar{G}_j(t) & \text{for } t < T, \\ 1 & \text{for } t \geq T. \end{cases}$$

The probability that the unit is replaced at time T is

$$\bar{F}(T) \prod_{j=1}^n \bar{G}_j(T),$$

the probability that it is replaced at time Y_j is

$$\int_0^T \bar{F}(t) \left[\prod_{i=1, i \neq j}^n \bar{G}_i(t) \right] dG_j(t) \quad (j = 1, 2, \dots, n),$$

and the probability that it is replaced at failure is

$$\int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] dF(t).$$

Thus, the mean time to replacement is

$$\begin{aligned} T\bar{F}(T) \left[\prod_{j=1}^n \bar{G}_j(T) \right] \\ + \sum_{j=1}^n \int_0^T t\bar{F}(t) \left[\prod_{i=1, i \neq j}^n \bar{G}_i(t) \right] dG_j(t) \\ + \int_0^T t \left[\prod_{j=1}^n \bar{G}_j(t) \right] dF(t) = \int_0^T \bar{F}(t) \left[\prod_{j=1}^n \bar{G}_j(t) \right] dt. \end{aligned}$$

Therefore, the expected cost rate is

$$C_F(T) = \frac{c_T + \sum_{j=1}^n (c_j - c_T) \int_0^T \bar{F}(t) \times \left[\prod_{i=1, i \neq j}^n \bar{G}_i(t) \right] dG_j(t) + (c_F - c_T) \int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] dF(t)}{\int_0^T \bar{F}(t) \left[\prod_{j=1}^n \bar{G}_j(t) \right] dt}, \quad (28.110)$$

where $c_j =$ replacement cost at time Y_j ($j = 1, 2, \dots, n$) and $c_F > c_j$.

We find optimal T_F^* to minimize $C_F(T)$ when $c_j = c_T$. Differentiating $C_F(T)$ with respect to T and setting it equal to zero,

$$h(T) \int_0^T \bar{F}(t) \left[\prod_{j=1}^n \bar{G}_j(t) \right] dt - \int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] dF(t) = \frac{c_T}{c_F - c_T}, \quad (28.111)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique T_F^* ($0 < T_F^* < \infty$) which satisfies (28.111), and the resulting cost rate is

$$C_F(T_F^*) = (c_F - c_T)h(T_F^*). \quad (28.112)$$

In addition, because the left-hand side of (28.111) decreases with n , T_F^* increases with n .

(2) Replacement Last

Suppose that the unit is replaced preventively at time T ($0 \leq T < \infty$) or at time Y_1, Y_2, \dots, Y_n ($n = 1, 2, \dots$), whichever occurs last. Thus, by setting $Z_M \equiv \text{Max}\{T, Y_1, \dots, Y_n\}$, Z_M has a distribution

$$G_L(t) \equiv \Pr\{Z_M \leq t\} = \begin{cases} 0 & \text{for } t < T, \\ \prod_{j=1}^n G_j(t) & \text{for } t \geq T. \end{cases}$$

The probability that the unit is replaced at time T is

$$\bar{F}(T) \prod_{j=1}^n G_j(T),$$

the probability that it is replaced at time Y_j is

$$\int_T^\infty \bar{F}(t) \left[\prod_{i=1, i \neq j}^n G_i(t) \right] dG_j(t) \quad (j = 1, 2, \dots, n),$$

and the probability that it is replaced at failure is

$$F(T) + \int_T^\infty \left[1 - \prod_{j=1}^n G_j(t) \right] dF(t).$$

Thus, the mean time to replacement is

$$\begin{aligned} & T\bar{F}(T) \prod_{j=1}^n G_j(T) \\ & + \sum_{j=1}^n \int_T^\infty t\bar{F}(t) \left[\prod_{i=1, i \neq j}^n G_i(t) \right] dG_j(t) \\ & + \int_0^T t dF(t) + \int_T^\infty t \left[1 - \prod_{j=1}^n G_j(t) \right] dF(t) \\ & = \int_0^T \bar{F}(t) dt + \int_T^\infty \bar{F}(t) \left[1 - \prod_{j=1}^n G_j(t) \right] dt. \end{aligned}$$

Therefore, the expected cost rate is

$$\begin{aligned} C_L(T) = & \\ & \frac{c_T + \sum_{j=1}^n (c_j - c_T) \int_T^\infty \bar{F}(t) \times \left[\prod_{i=1, i \neq j}^n G_i(t) \right] dG_j(t) + (c_F - c_T) \left\{ F(T) + \int_T^\infty \left[1 - \prod_{j=1}^n G_j(t) \right] dF(t) \right\}}{\int_0^T \bar{F}(t) dt + \int_T^\infty \bar{F}(t) \left[1 - \prod_{j=1}^n G_j(t) \right] dt}. \end{aligned} \quad (28.113)$$

We find optimal T_L^* to minimize $C_L(T)$ when $c_j = c_T$. Differentiating $C_L(T)$ with respect to T and setting it equal to zero,

$$\begin{aligned} & h(T) \left\{ \int_0^T \bar{F}(t) dt + \int_T^\infty \bar{F}(t) \left[1 - \prod_{j=1}^n G_j(t) \right] dt \right\} \\ & - F(T) - \int_T^\infty \left[1 - \prod_{j=1}^n G_j(t) \right] dF(t) = \frac{c_T}{c_F - c_T}, \end{aligned} \quad (28.114)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_L^* ($0 < T_L^* < \infty$) which satisfies (28.114), and the resulting cost rate is

$$C_L(T_L^*) = (c_F - c_T)h(T_L^*). \quad (28.115)$$

In addition, because the left-hand side of (28.114) decreases with n , T_L^* increases with n .

28.5.2 Periodic Replacement

(1) Replacement First

Suppose that the unit is replaced at time T ($0 < T \leq \infty$) or at time Y_1, Y_2, \dots, Y_n ($n = 1, 2, \dots$), whichever occurs first. Then, the probability that the unit is replaced at time T is

$$\prod_{j=1}^n \bar{G}_j(T),$$

and the probability that it is replaced at time Y_j is

$$\int_0^T \left[\prod_{i=1, i \neq j}^n \bar{G}_i(t) \right] dG_j(t) \quad (j = 1, 2, \dots, n).$$

Thus, the mean time to replacement is

$$\begin{aligned} & T \prod_{j=1}^n \bar{G}_j(T) + \sum_{j=1}^n \int_0^T t \left[\prod_{i=1, i \neq j}^n \bar{G}_i(t) \right] dG_j(t) \\ & = \int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] dt, \end{aligned}$$

and the expected number of failures is

$$\begin{aligned} & H(T) \prod_{j=1}^n \bar{G}_j(T) \\ & + \sum_{j=1}^n \int_0^T H(t) \left[\prod_{i=j, i \neq j}^n \bar{G}_i(t) \right] dG_j(t) \\ & = \int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] h(t) dt. \end{aligned}$$

Therefore, the expected cost rate is

$$\begin{aligned} C_F(T) = & \\ & \frac{c_T + \sum_{j=1}^n (c_j - c_T) \int_0^T \left[\prod_{i=1, i \neq j}^n \bar{G}_i(t) \right] dG_j(t) + c_M \int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] h(t) dt}{\int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] dt}. \end{aligned} \quad (28.116)$$

We find optimal T_F^* to minimize $C_F(T)$ when $c_j = c_T$. Differentiating $C_F(T)$ with respect to T and setting it equal to zero,

$$\begin{aligned} & h(T) \int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] dt - \int_0^T \left[\prod_{j=1}^n \bar{G}_j(t) \right] h(t) dt \\ & = \frac{c_T}{c_M}, \end{aligned} \quad (28.117)$$

whose left-hand side increases strictly with T from 0 to ∞ . Thus, there exists a finite and unique minimum T_F^* ($0 < T_F^* < \infty$) which satisfies (28.117), and the resulting cost rate is

$$C_F(T_F^*) = c_M h(T_F^*).$$

In addition, because the left-hand side of (28.117) decreases with n , T_F^* increases with n .

(2) Replacement Last

Suppose that the unit is replaced at time T ($0 \leq T < \infty$) or at time Y_1, Y_2, \dots, Y_n ($n = 1, 2, \dots$), whichever occurs last. Then, the probability that the unit is replaced at time T is

$$\prod_{j=1}^n G_j(T),$$

and the probability that it is replaced at time Y_j is

$$\int_T^\infty \left[\prod_{i=1, i \neq j}^n G_i(t) \right] dG_j(t) \quad (j = 1, 2, \dots, n).$$

Thus, the mean time to replacement is

$$\begin{aligned} T \prod_{j=1}^n G_j(T) + \sum_{j=1}^n \int_T^\infty t \left[\prod_{i=1, i \neq j}^n G_i(t) \right] dG_j(t) \\ = T + \int_T^\infty \left[1 - \prod_{j=1}^n G_j(t) \right] dt, \end{aligned}$$

and the expected number of failures until replacement is

$$\begin{aligned} H(T) \prod_{j=1}^n G_j(T) + \sum_{j=1}^n \int_T^\infty H(t) \left[\prod_{i=1, i \neq j}^n G_i(t) \right] dG_j(t) \\ = H(T) + \int_T^\infty \left[1 - \prod_{j=1}^n G_j(t) \right] h(t) dt. \end{aligned}$$

Therefore, the expected cost rate is

$$\begin{aligned} C_L(T) = \\ \frac{c_T + \sum_{j=1}^n (c_j - c_T) \int_T^\infty \left[\prod_{i=1, i \neq j}^n G_i(t) \right] dG_j(t) + c_M \{ H(T) + \int_T^\infty [1 - \prod_{j=1}^n G_j(t)] h(t) dt \}}{T + \int_T^\infty [1 - \prod_{j=1}^n G_j(t)] dt}. \end{aligned} \quad (28.118)$$

We find optimal T_L^* to minimize $C_L(T)$. Differentiating $C_L(T)$ with respect to T and setting it equal to zero,

$$\begin{aligned} h(T) \left\{ T + \int_T^\infty \left[1 - \prod_{j=1}^n G_j(t) \right] dt \right\} - H(T) \\ - \int_T^\infty \left[1 - \prod_{j=1}^n G_j(t) \right] h(t) dt = \frac{c_T}{c_M}, \end{aligned} \quad (28.119)$$

whose left-hand side increases strictly with T to ∞ . Thus, there exists a finite and unique T_L^* ($0 < T_L^* < \infty$) which satisfies (28.118), and the resulting cost rate is

$$C_L(T_L^*) = c_M h(T_L^*). \quad (28.120)$$

In addition, because the left-hand side of (28.119) increases with n , T_L^* decreases with n .

Appendix 1

It is assumed that the failure rate $h(t) \equiv f(t)/\bar{F}(t)$ increases strictly with t from $h(0)$ to $h(\infty)$, and for $N = 0, 1, 2, \dots$ and $K = 0, 1, 2, \dots$

$$\begin{aligned} g^{(N+1)}(t) &= \frac{\theta(\theta t)^N}{N!} e^{-\theta t}, \\ G^{(N)}(t) &= \int_0^t g^{(N)}(u) du = \sum_{j=N}^\infty \frac{(\theta t)^j}{j!} e^{-\theta t}, \\ p_K(t) &= \frac{H(t)^K}{K!} e^{-H(t)}, \\ P_K(t) &= \sum_{j=K}^\infty p_j(t) = \sum_{j=K}^\infty \frac{H(t)^j}{j!} e^{-H(t)}. \end{aligned}$$

When $F(t) = 1 - e^{-H(t)}$, for $0 < T < \infty$,

$$\begin{aligned} Q(T) &= \frac{\bar{F}(T)}{\int_T^\infty \bar{F}(t) dt}, \quad Q_O(T) = \frac{\int_0^T e^{-\theta t} dF(t)}{\int_0^T e^{-\theta t} \bar{F}(t) dt}, \\ \tilde{Q}_O(T) &= \frac{\int_T^\infty e^{-\theta t} dF(t)}{\int_T^\infty e^{-\theta t} \bar{F}(t) dt}, \end{aligned}$$

all of which increase strictly with T , and for $0 < T < \infty$,

$$\begin{aligned} Q(\infty) &= \tilde{Q}_O(\infty) = h(\infty), \\ Q_O(T) &< h(T) < \tilde{Q}_O(T) < Q(T). \end{aligned}$$

Appendix 2

1. For $N = 0, 1, 2, \dots$ and $0 < T \leq \infty$,

$$Q(T, N) \equiv \frac{\int_0^T (\theta t)^N e^{-\theta t} dF(t)}{\int_0^T (\theta t)^N e^{-\theta t} \bar{F}(t) dt}$$

increases strictly with T from $h(0)$ to $Q(N) \equiv \int_0^\infty (\theta t)^N e^{-\theta t} dF(t) / \int_0^\infty (\theta t)^N e^{-\theta t} \bar{F}(t) dt$ and increases strictly with N from $Q_O(T)$ to $h(T)$.

2. For $N = 0, 1, 2, \dots$ and $0 \leq T < \infty$,

$$\tilde{Q}(T, N) \equiv \frac{\int_T^\infty (\theta t)^N e^{-\theta t} dF(t)}{\int_T^\infty (\theta t)^N e^{-\theta t} \bar{F}(t) dt}$$

increases strictly with T from $Q(N)$ to $h(\infty)$ and increases strictly with N from $\tilde{Q}_O(T)$ to $h(\infty)$.

Proof. Note that

$$\begin{aligned} Q(0, N) &= h(0), & Q(\infty, N) &= Q(N), \\ Q(T, 0) &= Q_0(T), & Q(T, \infty) &= h(T), \\ \tilde{Q}(0, N) &= Q(N), & \tilde{Q}(\infty, N) &= h(\infty), \\ \tilde{Q}(T, 0) &= \tilde{Q}_0(T), & \tilde{Q}(T, \infty) &= h(\infty). \end{aligned}$$

Differentiating $Q(T, N)$ with respect to T ,

$$(\theta T)^N e^{-\theta T} \bar{F}(T) \int_0^T (\theta t)^N e^{-\theta t} [h(T) - h(t)] dt > 0,$$

which follows that $Q(T, N)$ increases strictly with T from $h(0)$ to $Q(N)$ for any N . Forming $Q(T, N + 1) - Q(T, N)$ and denoting

$$\begin{aligned} L_1(T) &\equiv \\ &\int_0^T (\theta t)^{N+1} e^{-\theta t} dF(t) \int_0^T (\theta t)^N e^{-\theta t} \bar{F}(t) dt \\ &\quad - \int_0^T (\theta t)^N e^{-\theta t} dF(t) \int_0^T (\theta t)^{N+1} e^{-\theta t} \bar{F}(t) dt, \end{aligned}$$

we have $L_1(0) = 0$ and

$$\begin{aligned} L'_1(T) &= (\theta T)^N e^{-\theta T} \bar{F}(T) \\ &\quad \times \int_0^T (\theta t)^N e^{-\theta t} \bar{F}(t) (\theta T - \theta t) [h(T) - h(t)] dt > 0, \end{aligned}$$

which follows that $Q(T, N)$ increases strictly with N from $Q_0(T)$ to $h(T)$ for any T .

Using the similar method, we can easily prove 2. Therefore, taking $T \rightarrow \infty$, $Q(N)$ increases strictly with N to $h(\infty)$, and for $0 < T, N < \infty$,

$$\begin{aligned} Q(T, N) &< h(T) < \tilde{Q}_0(T) < \tilde{Q}(T, N), \\ Q(T, N) &< Q(N) < \tilde{Q}(T, N). \end{aligned}$$

Appendix 3

1. For $N = 0, 1, 2, \dots$ and $0 < T \leq \infty$,

$$H(T, N) \equiv \frac{\int_0^T (\theta t)^N e^{-\theta t} h(t) dt}{\int_0^T (\theta t)^N e^{-\theta t} dt}$$

increases strictly with T from $h(0)$ to $H(N) \equiv \int_0^\infty [\theta(\theta t)^N / N!] e^{-\theta t} h(t) dt$ and increases strictly with N from $H(T, 0) = \int_0^T \theta e^{-\theta t} h(t) dt / (1 - e^{-\theta T})$ to $h(T)$.

2. For $N = 0, 1, 2, \dots$ and $0 \leq T < \infty$,

$$\tilde{H}(T, N) \equiv \frac{\int_T^\infty (\theta t)^N e^{-\theta t} h(t) dt}{\int_T^\infty (\theta t)^N e^{-\theta t} dt}$$

increases strictly with T from $H(N)$ to $h(\infty)$ and increases strictly with N from $\int_0^\infty \theta e^{-\theta t} h(T + t) dt$ to $h(\infty)$.

Proof. Note that

$$\begin{aligned} H(0, N) &= h(0), & H(\infty, N) &= H(N), \\ H(T, 0) &= \frac{\int_0^T \theta e^{-\theta t} h(t) dt}{1 - e^{-\theta T}}, & H(T, \infty) &= h(T), \\ \tilde{H}(0, N) &= H(N), & \tilde{H}(\infty, N) &= h(\infty), \\ \tilde{H}(T, 0) &= \int_0^\infty \theta e^{-\theta t} h(T + t) dt, & \tilde{H}(T, \infty) &= h(\infty). \end{aligned}$$

Using the similar method in Appendix 1, we can prove 1 and 2. Therefore, for $0 < T, N < \infty$,

$$\begin{aligned} H(T, N) &< h(T) < \tilde{H}(T, N), \\ H(T, N) &< H(N) < \tilde{H}(T, N), \end{aligned}$$

and $H(N)$ increases strictly with N from $\int_0^\infty \theta e^{-\theta t} h(t) dt$ to $h(\infty)$.

Appendix 4

1. For $N = 0, 1, 2, \dots$ and $0 < T \leq \infty$,

$$Q_1(T, N) \equiv \frac{\int_0^T (\theta t)^N [\int_t^\infty e^{-\theta u} dF(u)] dt}{\int_0^T (\theta t)^N [\int_t^\infty e^{-\theta u} \bar{F}(u) du] dt}$$

increases strictly with T from $\tilde{Q}_0(0) = \int_0^\infty e^{-\theta t} dF(t) / \int_0^\infty e^{-\theta t} \bar{F}(t) dt$ to $Q(N + 1)$ and increases strictly with N from $Q_1(T, 0)$ to $\tilde{Q}_0(T)$.

2. For $N = 0, 1, 2, \dots$ and $0 \leq T < \infty$,

$$\tilde{Q}_1(T, N) \equiv \frac{\int_T^\infty (\theta t)^N [\int_t^\infty e^{-\theta u} dF(u)] dt}{\int_T^\infty (\theta t)^N [\int_t^\infty e^{-\theta u} \bar{F}(u) du] dt}$$

increases strictly with T from $Q(N + 1)$ to $h(\infty)$ and increases strictly with N from $\tilde{Q}_1(T, 0)$ to $h(\infty)$.

Proof. Note that

$$Q_1(0, N) = \frac{\int_0^\infty e^{-\theta t} dF(t)}{\int_0^\infty e^{-\theta t} \bar{F}(t) dt} = \tilde{Q}_0(0),$$

$$\begin{aligned} Q_1(\infty, N) &= Q(N + 1), & Q_1(T, \infty) &= \tilde{Q}_O(T), \\ \tilde{Q}_1(0, N) &= Q(N + 1), & \tilde{Q}_1(\infty, N) &= h(\infty), \\ \tilde{Q}_1(T, \infty) &= h(\infty). \end{aligned}$$

Differentiating $Q_1(T, N)$ with respect to T ,

$$\begin{aligned} &(\theta T)^N \int_T^\infty e^{-\theta t} \bar{F}(t) dt \int_0^T (\theta t)^N \\ &\times \left[\int_t^\infty e^{-\theta u} \bar{F}(u) du \right] [\tilde{Q}_O(T) - \tilde{Q}_O(t)] dt > 0, \end{aligned}$$

which follows that $Q_1(T, N)$ increases strictly with T from $\tilde{Q}_O(0)$ to $Q(N + 1)$ for any N . Forming $Q_1(T, N + 1) - Q_1(T, N)$ and denoting

$$\begin{aligned} L_3(T) &= \int_0^T (\theta t)^{N+1} \left[\int_t^\infty e^{-\theta u} dF(u) \right] dt \\ &\times \int_0^T (\theta t)^N \left[\int_t^\infty e^{-\theta u} \bar{F}(u) du \right] dt \\ &- \int_0^T (\theta t)^N \left[\int_t^\infty e^{-\theta u} dF(u) \right] dt \\ &\times \int_0^T (\theta t)^{N+1} \left[\int_t^\infty e^{-\theta u} \bar{F}(u) du \right] dt, \end{aligned}$$

we have $L_3(0) = 0$ and

$$\begin{aligned} L'_3(T) &= (\theta T)^N \left[\int_T^\infty e^{-\theta t} \bar{F}(t) dt \right] \\ &\times \int_0^T (\theta t)^N \left[\int_t^\infty e^{-\theta u} \bar{F}(u) du \right] \\ &\times (\theta T - \theta t) [\tilde{Q}_O(T) - \tilde{Q}_O(t)] dt > 0, \end{aligned}$$

which follows that $Q_1(T, N)$ increases strictly with N from $Q_1(T, 0)$ to $\tilde{Q}_O(T)$ for any T .

Using the similar method, we can prove 2, and for $0 < T, N < \infty$,

$$\begin{aligned} Q_1(T, N) &< \tilde{Q}_O(T) < \tilde{Q}_1(T, N), \\ Q_1(T, N) &< Q(N + 1) < \tilde{Q}_1(T, N). \end{aligned}$$

Appendix 5

1. For $K = 0, 1, 2, \dots$ and $0 < T \leq \infty$,

$$H(T, K) \equiv \frac{\int_0^T p_K(t) h(t) dt}{\int_0^T p_K(t) dt}$$

increases strictly with T from $h(0)$ to $H(\infty, K) = 1 / \int_0^\infty p_K(t) dt$ and increases strictly with K from $F(T) / \int_0^T \bar{F}(t) dt$ to $h(T)$.

2. For $K = 0, 1, 2, \dots$ and $0 \leq T < \infty$,

$$\tilde{H}(T, K) \equiv \frac{\int_T^\infty p_K(t) h(t) dt}{\int_T^\infty p_K(t) dt}$$

increases strictly with T from $1 / \int_0^\infty p_K(t) dt$ to $h(\infty)$ and increases strictly with K from $Q(T)$ to $h(\infty)$.

Proof. Note that

$$\begin{aligned} H(0, K) &= h(0), & H(\infty, K) &= \frac{1}{\int_0^\infty p_K(t) dt}, \\ H(T, 0) &= \frac{F(T)}{\int_0^T \bar{F}(t) dt}, & H(T, \infty) &= h(T), \\ \tilde{H}(0, K) &= \frac{1}{\int_0^\infty p_K(t) dt}, & \tilde{H}(\infty, K) &= h(\infty), \\ \tilde{H}(T, 0) &= Q(T), & \tilde{H}(T, \infty) &= h(\infty). \end{aligned}$$

Differentiating $H(T, K)$ with respect to T

$$p_K(T) \int_0^T p_K(t) [h(T) - h(t)] dt > 0,$$

which follows that $H(T, K)$ increases strictly with T from $h(0)$ to $1 / \int_0^\infty p_K(t) dt$ for any K . Forming $H(T, K + 1) - H(T, K)$ and denoting

$$\begin{aligned} L_4(T) &\equiv \int_0^T p_{K+1}(t) h(t) dt \int_0^T p_K(t) dt \\ &- \int_0^T p_K(t) h(t) dt \int_0^T p_{K+1}(t) dt, \end{aligned}$$

we have $L_4(0) = 0$ and

$$\begin{aligned} L'_4(T) &= \frac{p_K(T) h(T)}{(K + 1)} \\ &\times \int_0^T p_K(t) [H(T) - H(t)] [h(T) - h(t)] dt > 0, \end{aligned}$$

which follows that $H(T, K)$ increases strictly with K from $F(T) / \int_0^T \bar{F}(t) dt$ to $h(T)$ for any T .

Using the similar method, we can prove 2. Therefore, for $0 \leq K < \infty$ and $0 < T < \infty$,

$$\begin{aligned} H(T, K) &< h(T) < \tilde{H}(T, K), \\ H(T, K) &< \frac{1}{\int_0^\infty p_K(t) dt} < \tilde{H}(T, K), \end{aligned}$$

and $1 / \int_0^\infty p_K(t) dt$ increases strictly with K from $1 / \int_0^\infty p_0(t) dt = 1 / \mu$ to $h(\infty)$.

Appendix 6

1. For $K = 0, 1, 2, \dots$ and $0 < T \leq \infty$,

$$H_1(T, K) \equiv \frac{P_K(T)}{\int_0^T [\int_t^\infty e^{-H(u)+H(t)} du] dP_K(t)}$$

increases strictly with T from $1/\mu$ to $1/\int_0^\infty p_K(t)dt$ and increases strictly with K from $1/\mu$ to $Q(T)$.

2. For $K = 1, 2, \dots$ and $0 \leq T < \infty$,

$$\tilde{H}_1(T, K) \equiv \frac{\bar{P}_K(T)}{\int_T^\infty [\int_t^\infty e^{-H(u)+H(t)} du] dP_K(t)}$$

increases strictly with T from $1/\int_0^\infty p_K(t)dt$ to $h(\infty)$ and increases strictly with K from $\tilde{H}_1(T, 1)$ to $h(\infty)$.

Proof. Note that

$$\begin{aligned} H_1(0, K) &= \frac{1}{\mu}, \\ H_1(\infty, K) &= \frac{1}{\int_0^\infty [\int_t^\infty e^{-H(u)+H(t)} du] dP_K(t)} \\ &= \frac{1}{\int_0^\infty p_K(t)dt}, \\ H_1(T, \infty) &= \frac{\bar{F}(T)}{\int_T^\infty \bar{F}(T)dt} = Q(T), \\ \tilde{H}_1(0, K) &= \frac{1}{\int_0^\infty p_K(t)dt}, \\ \tilde{H}_1(\infty, K) &= h(\infty), \quad \tilde{H}_1(T, \infty) = h(\infty). \end{aligned}$$

Using the similar method in Appendix 4, we can prove 1 and 2. Therefore, for $0 < T, K < \infty$,

$$\begin{aligned} H_1(T, K) &< Q(T) < \tilde{H}_1(T, K), \\ \frac{1}{\mu} &< H_1(T, K) < \frac{1}{\int_0^\infty p_K(t)dt} < \tilde{H}_1(T, K). \end{aligned}$$

Appendix 7

1. For $N = 0, 1, 2, \dots$ and $0 < T \leq \infty$,

$$H_1(T, N) \equiv \frac{\int_0^T [\int_0^\infty \theta e^{-\theta u} h(t+u) du] dG^{(N)}(t)}{G^{(N)}(T)}$$

increases strictly with T from $\int_0^\infty \theta e^{-\theta t} h(t)dt$ to $H(N)$ and increases strictly with N from $\int_0^\infty \theta e^{-\theta t} h(t)dt$ to $\int_0^\infty \theta e^{-\theta t} h(T+t)dt$.

2. For $N = 1, 2, \dots$ and $0 \leq T < \infty$,

$$\begin{aligned} \tilde{H}_1(T, N) &\equiv \\ &\frac{\int_T^\infty [\int_0^\infty \theta e^{-\theta u} h(t+u) du] dG^{(N)}(t)}{1 - G^{(N)}(T)} \end{aligned}$$

increases strictly with T from $H(N)$ to $h(\infty)$ and increases strictly with N from $\tilde{H}_1(T, 1)$ to $h(\infty)$.

Proof. Note that

$$\begin{aligned} H_1(0, N) &= \int_0^\infty \theta e^{-\theta t} h(t)dt, \\ H_1(\infty, N) &= \int_0^\infty \left[\int_0^\infty \theta e^{-\theta u} h(t+u) du \right] dG^{(N)}(t) \\ &= H(N), \\ H_1(T, 0) &= \int_0^\infty \theta e^{-\theta t} h(t)dt, \\ H_1(T, \infty) &= \int_0^\infty e^{-\theta t} h(T+t)dt, \\ \tilde{H}_1(0, N) &= H(N), \quad \tilde{H}_1(\infty, N) = h(\infty), \\ \tilde{H}_1(T, \infty) &= h(\infty). \end{aligned}$$

Differentiating $H_1(T, N)$ with respect to T ,

$$\begin{aligned} g^{(N)}(T) \int_0^T \left[\int_0^\infty \theta e^{-\theta u} h(T+u) du \right. \\ \left. - \int_0^\infty \theta e^{-\theta u} h(t+u) du \right] dG^{(N)}(t) > 0, \end{aligned}$$

which follows that $H_1(T, N)$ increases strictly with T from $\int_0^\infty \theta e^{-\theta t} h(t)dt$ to $H(N)$ for any N . Forming $H_1(T, N+1) - H_1(T, N)$ and denoting

$$\begin{aligned} L_5(T) &\equiv \\ &G^{(N)}(T) \int_0^T \left[\int_0^\infty \theta e^{-\theta u} h(t+u) du \right] dG^{(N+1)}(t) \\ &- G^{(N+1)}(T) \int_0^T \left[\int_0^\infty \theta e^{-\theta u} h(t+u) du \right] dG^{(N)}(t), \end{aligned}$$

we have $L_5(0) = 0$ and

$$\begin{aligned} L_5'(T) &= g^{(N)}(T) \int_0^T \left[\int_0^\infty \theta e^{-\theta u} h(T+u) du \right. \\ &\left. - \int_0^\infty \theta e^{-\theta u} h(t+u) du \right] (\theta T - \theta t) dG^{(N)}(t) > 0, \end{aligned}$$

which follows that $H_1(T, N)$ increases strictly with N from $H_1(T, 1)$ to $\int_0^\infty \theta e^{-\theta t} h(T+t)dt$ for any T .

Using the similar method, we can easily prove 2. Therefore, for $0 < T, N < \infty$,

$$H_1(T, N) < \int_0^\infty \theta e^{-\theta t} h(T+t) dt < \tilde{H}_1(T, N),$$

$$H_1(T, N) < H(N) < \tilde{H}_1(T, N).$$

Appendix 8

1. For $N = 1, 2, \dots$ and $K = 0, 1, 2, \dots$,

$$H_1(K, N) \equiv \frac{\int_0^\infty [1 - G^{(N)}(t)] p_K(t) h(t) dt}{\int_0^\infty [1 - G^{(N)}(t)] p_K(t) dt}$$

increases strictly with K from $H(0, N)$ to $h(\infty)$ and increases strictly with N from $H(K, 1)$ to $1 / \int_0^\infty p_K(t) dt$.

2. For $N = 0, 1, 2, \dots$ and $K = 1, 2, \dots$,

$$H_2(K, N) \equiv \frac{\int_0^\infty (\theta t)^N e^{-\theta t} \bar{P}_K(t) h(t) dt}{\int_0^\infty (\theta t)^N e^{-\theta t} \bar{P}_K(t) dt}$$

increases strictly with K from $H_2(1, N)$ to $H(N)$ and increases strictly with N from $H_2(K, 0)$ to $h(\infty)$.

Proof. Denoting that for $0 < T \leq \infty$,

$$H_1(T; K, N) \equiv \frac{\int_0^T [1 - G^{(N)}(t)] p_K(t) h(t) dt}{\int_0^T [1 - G^{(N)}(t)] p_K(t) dt},$$

we have

$$\lim_{K \rightarrow \infty} H_1(T; K, N) = h(T),$$

$$\lim_{N \rightarrow \infty} H(T; K, N) = \frac{\int_0^T p_K(t) h(t) dt}{\int_0^T p_K(t) dt} = H(T, K).$$

Furthermore, using the similar methods in Appendixes 3 and 5, $H_1(T; K, N)$ increases strictly with K and increases strictly with N . Thus, $H_1(T; K, N)$ increases strictly with K to $h(T)$ and increases strictly with N to $H(T, K)$ for any T . Taking $T \rightarrow \infty$, $H_1(K, N)$ increases strictly with K to $h(\infty)$ for any N and increases strictly with N to $1 / \int_0^\infty p_K(t) dt$ for any N .

Using the similar method, we can prove 2.

Appendix 9

1. For $N = 0, 1, 2, \dots$ and $K = 0, 1, 2, \dots$,

$$H_3(K, N) \equiv \frac{\int_0^\infty G^{(N)}(t) p_K(t) h(t) dt}{\int_0^\infty G^{(N)}(t) p_K(t) dt}$$

increases strictly with K from $H_3(0, N)$ to $h(\infty)$ and increases strictly with N from $1 / \int_0^\infty p_K(t) dt$ to $h(\infty)$.

2. For $N = 0, 1, 2, \dots$ and $K = 0, 1, 2, \dots$,

$$H_4(K, N) \equiv \frac{\int_0^\infty (\theta t)^N e^{-\theta t} P_K(t) h(t) dt}{\int_0^\infty (\theta t)^N e^{-\theta t} P_K(t) dt}$$

increases strictly with K from $H(N)$ to $h(\infty)$ and increases strictly with N from $H_4(K, 0)$ to $h(\infty)$. Therefore, for $0 < K, N < \infty$,

$$H_1(K, N) < \frac{1}{\int_0^\infty p_K(t) dt} < H_3(K, N),$$

$$H_2(K, N) < H(N) < H_4(K, N).$$

Proof. Using the similar method in Appendix 8, we can prove 1 and 2.

Appendix 10

For $K = 0, 1, 2, \dots$ and $0 \leq T \leq \infty$,

$$H_{M1}(T, K) \equiv \frac{\int_0^\infty h(T+t) p_K(t) dt}{\int_0^\infty p_K(t) dt}$$

increases strictly with T from $1 / \int_0^\infty p_K(t) dt$ to $h(\infty)$ and increases strictly with K from $H_{M1}(T, 0)$ to $h(\infty)$.

Proof. It can be clearly shown that $H_{M1}(T, K)$ increases strictly with T from $1 / \int_0^\infty p_K(t) dt$ to $h(\infty)$ for any K . Next, denoting that for $0 < T_1 < \infty$,

$$H(T_1, T, K) \equiv \frac{\int_0^{T_1} h(T+t) p_K(t) dt}{\int_0^{T_1} p_K(t) dt},$$

we have $\lim_{K \rightarrow \infty} H(T_1, T, K) = h(T + T_1)$. Forming $H(T_1, T, K + 1) - H(T_1, T, K)$ and denoting

$$L_6(T_1) \equiv \int_0^{T_1} h(T+t) p_{K+1}(t) dt \int_0^{T_1} p_K(t) dt$$

$$- \int_0^{T_1} h(T+t) p_K(t) dt \int_0^{T_1} p_{K+1}(t) dt,$$

we have $L_6(0) = 0$ and

$$L'_6(T_1) = \frac{p_K(T_1)}{K+1} \int_0^{T_1} p_K(t)[H(T_1) - H(t)] \times [h(T+T_1) - h(T+t)]dt > 0,$$

which follows that $H(T_1, T, K)$ increases strictly with K to $h(T+T_1)$ for any T_1 . Taking $T_1 \rightarrow \infty$, $H_{M1}(T, K)$ increases strictly with K to $h(\infty)$ for any T , and $H_{M1}(T, K) \geq h(T)$ and $H_{M1}(T, K) \geq 1/\int_0^\infty p_K(t)dt$.

Appendix 11

For $K = 0, 1, 2, \dots$ and $0 \leq T < \infty$,

$$H_{M2}(T, K) \equiv \frac{1}{\int_T^\infty p_K(t, T)dt}$$

increases strictly with T from $1/\int_0^\infty p_K(t)dt$ to $h(\infty)$ and increases strictly with K from $Q(T)$ to $h(\infty)$.

Proof. Note first that

$$H_{M2}(T, K) = \frac{\int_T^\infty [H(t) - H(T)]^K e^{-H(t)} h(t) dt}{\int_T^\infty [H(t) - H(T)]^K e^{-H(t)} dt},$$

because

$$\int_T^\infty [H(t) - H(T)]^K e^{-H(t)} h(t) dt = K! e^{-H(T)}.$$

Next, note that

$$H_{M2}(0, K) = \frac{1}{\int_0^\infty p_K(t)dt}, \quad H_{M2}(\infty, K) = h(\infty),$$

$$H_{M2}(T, 0) = Q(T), \quad H_{M2}(T, \infty) = h(\infty).$$

Differentiating $H_{M2}(T, K)$ with respect to T ,

$$Kh(T) \left\{ \int_T^\infty [H(t) - H(T)]^K dF(t) \times \int_T^\infty [H(u) - H(T)]^{K-1} \bar{F}(u) du - \int_T^\infty [H(t) - H(T)]^{K-1} dF(t) \times \int_T^\infty [H(u) - H(T)]^K \bar{F}(u) du \right\} = Kh(T) \int_T^\infty [H(t) - H(T)]^{K-1} \bar{F}(t) dt$$

$$\times \int_T^t [H(u) - H(T)]^{K-1} \bar{F}(u) \times [h(t) - h(u)][H(t) - H(u)] du > 0,$$

because

$$\int_T^\infty [H(t) - H(T)]^K dF(t) \times \left\{ \int_T^t [H(u) - H(T)]^{K-1} \bar{F}(u) du + \int_t^\infty [H(u) - H(T)]^{K-1} \bar{F}(u) du \right\} = \int_T^\infty [H(t) - H(T)]^K dF(t) \times \int_T^t [H(u) - H(T)]^{K-1} \bar{F}(u) du + \int_T^\infty [H(t) - H(T)]^{K-1} \bar{F}(t) dt \times \int_T^t [H(u) - H(T)]^K dF(u),$$

which follows that $H_{M2}(T, K)$ increases strictly with T from $1/\int_0^\infty p_K(t)dt$ to $h(\infty)$ for any K . Forming $H_{M2}(T, K+1) - H_{M2}(T, K)$, and denoting

$$L_7(T) \equiv \int_T^\infty [H(t) - H(T)]^{K+1} dF(t) \times \int_T^\infty [H(u) - H(T)]^K \bar{F}(u) du - \int_T^\infty [H(t) - H(T)]^K dF(t) \times \int_T^\infty [H(u) - H(T)]^{K+1} \bar{F}(u) du,$$

we have $L_7(\infty) = 0$ and $L'_7(T) < 0$ which follows that $H_{M2}(T, K)$ increases strictly with K from $Q(T)$ to $h(\infty)$ for any T , and $H_{M2}(T, K) \geq 1/\int_0^\infty p_K(t)dt$, $H_{M2}(T, K) \geq Q(T)$.

References

1. Barlow, R.E., Proschan, F.: Mathematical Theory of Reliability. Wiley, New York (1965)
2. Nakagawa, T.: Maintenance Theory of Reliability. Springer, Berlin (2005)

3. Nakagawa, T., Zhao, X.: Optimization problems of a parallel system with a random number of units. *IEEE Trans. Reliab.* **61**, 543–548 (2012)
4. Zhao, X., Nakagawa, T.: Optimization problems of replacement first or last in reliability theory. *Eur. J. Oper. Res.* **223**, 141–149 (2012)
5. Zhao, X., Al-Khalifa, K.N., Hamouda, A.M.G., Nakagawa, T.: What is middle maintenance policy? *Quality Reliab. Eng. Int.* **32**, 2403–2414 (2016)
6. Chen, M., Zhao, X., Nakagawa, T.: Replacement policies with general models. *Ann. Oper. Res.* (2017). <https://doi.org/10.1007/s10479-017-2685-y>
7. Nakagawa, T.: *Random Maintenance Policies*. Springer, London (2014)
8. Nakagawa, T., Zhao, X.: *Maintenance Overtime Policies in Reliability Theory*. Springer, Switzerland (2015)
9. Zhao, X., Al-Khalifa, K.N., Hamouda, A.M.S., Nakagawa, T.: First and last triggering event approaches for replacement with minimal repairs. *Trans. Reliab.* **65**, 197–207 (2016)
10. Mizutani, S., Zhao, X., Nakagawa, T.: Which replacement is better at working cycles or number of failures. *IEICE Trans. Fundamentals* **E103-A(2)**, 523–532 (2020)



Toshio Nakagawa received BSE and MS degrees from Nagoya Institute of Technology, and Doctor of Engineering degree from Kyoto University. He is now a Honorary Professor at Aichi Institute of Technology in Toyota City. He has published 7 books from Springer, 20 books chapters and 200 journal papers. His research interests are analysis for stochastic systems in reliability and maintenance theory.



Satoshi Mizutani is Associate Professor at Aichi Institute of Technology. He received Ph.D. degree from Aichi Institute of Technology. He was a visiting researcher at Kinjo Gakuin University in Nagoya City. He worked as Assistant Professor, and as Associate Professor at Aichi University of Technology. His research interests are optimal maintenance policy in reliability theory.




Xufeng Zhao is a Professor at Nanjing University of Aeronautics and Astronautics, China. He received his doctoral degree in business administration and computer science in 2013 from Aichi Institute of Technology, Japan. He is interested in probability theory, stochastic process, reliability and maintenance theory, and applications in computer and industrial systems.

Advanced Statistical Methods and Modeling



Confidence Distribution and Distribution Estimation for Modern Statistical Inference

29

Yifan Cui  and Min-ge Xie

Contents

29.1	Introduction	575
29.2	Confidence Distribution	576
29.2.1	The Concept of CD.....	576
29.2.2	CD-Based Inference.....	578
29.2.3	Combination of CDs for Fusion Learning.....	580
29.2.4	Multivariate CDs.....	580
29.3	Fiducial Inference	581
29.3.1	Fiducial Inference.....	581
29.3.2	Generalized Fiducial Distribution.....	581
29.3.3	A User-Friendly Formula for GFD.....	582
29.3.4	Examples of GFD.....	582
29.4	Applications and Numerical Examples	585
29.4.1	CD-Based Inference.....	585
29.4.2	Nonparametric GFD-Based Inference.....	587
29.4.3	Combining Information from Multiple CDs.....	589
References	589

Keywords

Confidence distributions · Fiducial inference · Distributional inference · Confidence intervals · Coverage · Fusion learning · Bayesian · fiducial · and frequentist (BFF)

29.1 Introduction

A confidence distribution (CD) refers to a sample-dependent distribution function that can represent confidence intervals (regions) of all levels for a parameter of interest [74, 90]. Instead of the usual point estimator or confidence interval, CD is a distribution estimator of a parameter of interest with a pure frequentist interpretation. The development of the CD can be traced back to, for example, [16, 28, 47, 66]. However, its associated inference schemes and applications have not received much attention until the recent surge of interest in the research of CD and its applications [25, 46, 52, 53, 72–74, 77, 78, 82, 90, 91, 94]. All of these developments of CDs, along with a modern definition and interpretation, provide a powerful inferential tool for statistical inference.

One of the main contributions of CD is its applications on fusion learning [12, 15, 40, 51–53, 72, 75, 77, 81, 91, 92]. Combining CDs from independent studies naturally preserves more information from the individual studies than a traditional approach of combining only point estimators. A unified framework of combining CDs for fusion learning generally includes three steps: (1) using a CD to summarize relevant information or obtain an inference result from each study, (2) combining information from different sources or studies by combining these CDs, and (3) making inference via the combined CD. This approach has sound theoretical support and has been applied to many practical situations with much success.

On a different note, the fiducial distribution may be considered as one special type of CD, which provides a

Abstract

This chapter introduces to readers the new concept and methodology of confidence distribution and the modern-day distributional inference in statistics. This discussion should be of interest to people who would like to go into the depth of the statistical inference methodology and to utilize distribution estimators in practice. We also include in the discussion the topic of generalized fiducial inference, a special type of modern distributional inference, and relate it to the concept of confidence distribution. Several real data examples are also provided for practitioners. We hope that the selected content covers the greater part of the developments on this subject.

Y. Cui (✉)
Zhejiang University, Hangzhou, Zhejiang, China
e-mail: cuiyf@zju.edu.cn

M. Xie
Rutgers University, New Brunswick, NJ, USA
e-mail: mxie@stat.rutgers.edu

systematic way to obtain a CD. The origin of fiducial inference can be traced back to R.A. Fisher [28] who introduced the concept of a fiducial distribution for one parameter and proposed the use of this fiducial distribution to avoid the problems related to the choice of a prior distribution. Since the mid-2000s, there has been a renewed interest in modifications of fiducial inference [2, 7, 8, 22, 24, 30, 31, 33–37, 41, 56, 57, 59–63, 68, 73, 74, 79, 85, 87, 90, 93, 96].

We briefly overview these modern approaches which extend Fisher's original fiducial argument. We then focus on a recent development termed generalized fiducial inference and its applications [14, 17, 37, 41, 42, 44, 49, 50, 65, 86, 88, 89] that greatly expand the applicability of fiducial ideas. We demonstrate this recipe on several examples of varying complexity. The statistical procedures derived by the generalized fiducial inference often have very good performance from both theoretical and numerical points of view.

29.2 Confidence Distribution

29.2.1 The Concept of CD

This section will mainly focus on the concept of CD. The CD can be viewed as a distribution estimator, which can be utilized for constructing statistical procedures such as point estimates, confidence intervals, hypothesis tests, etc. The basic notion of CDs is related to the fiducial distribution of [28]; however, it is a pure frequentist concept. Some have suggested to view CD as the frequentist analog of Bayesian posterior distribution [e.g., 73, 74]. More broadly, if the credible intervals or regions obtained from a Bayesian posterior match with frequentist intervals or regions (either exactly or asymptotically), then the Bayesian posterior can be viewed as CD, and thus Bayesian approach is also a way to obtain CD [90].

Suppose X_1, X_2, \dots, X_n are independent and identically distributed and \mathcal{X} is the sample space corresponding to the dataset (X_1, X_2, \dots, X_n) . Let θ be a scalar parameter of interest and Θ be the parameter space. The following formal definitions of CD and asymptotic CD are proposed in [72, 77].

Definition 29.2.1 (CD and Asymptotic CD) A function $H_n(\cdot) = H_n(x, \cdot)$ on $\mathcal{X} \times \Theta \rightarrow [0, 1]$ is called a CD for a parameter, if (1) for each given $x \in \mathcal{X}$, $H_n(\cdot)$ is a (continuous) cumulative distribution function on Θ and (2) at the true parameter value $\theta = \theta_0$, $H_n(\theta_0) \equiv H_n(x, \theta_0)$, as a function of the sample x , follows the uniform distribution $U(0, 1)$. In addition, the function $H_n(\cdot)$ is called an asymptotic CD if condition (2) is replaced by (2') at the true parameter $\theta = \theta_0$, $H_n(\theta_0) \xrightarrow{d} U(0, 1)$ as $n \rightarrow \infty$.

From a nontechnical point of view, a CD is a function of both the parameter and the sample which satisfies two conditions. The first condition basically states that for any fixed sample, a CD is a distribution function on the parameter space. The second condition essentially requires that the corresponding inference derived by a CD has desired frequentist properties. Section 29.2.2 will further discuss how to use the second condition to extract information from a CD to make inference.

Birnbaum [9] introduced the concept of confidence curve as "an omnibus technique for estimation and testing statistical hypotheses," which was independent of the development of CD. From a CD $H_n(\theta)$, the confidence curve can be written as

$$CV_n(\theta) = 2 \min\{H_n(\theta), 1 - H_n(\theta)\}.$$

Indeed, confidence curve is an alternative expression of CD and it is a very useful graphical tool for visualizing CDs. On a plot of $CV_n(\theta)$ versus θ , a line across the y -axis of the significance level α , for any $0 < \alpha < 1$, intersects with the confidence curve at two points, and these two points correspond to an $1 - \alpha$ level, equal-tailed, two-sided confidence interval for θ . In addition, the maximum of a confidence curve is the median of the CD which is the recommended point estimator.

We present below five illustrating examples of CDs. More examples refer to [74, 77, 90].

Example 29.2.1 Suppose the data $X_i \sim N(\mu, 1)$, $i = 1, \dots, n$, with unknown μ . Let \bar{x}_n denote the sample mean. Then $N(\bar{x}_n, 1/n)$ is a CD for μ , and it can be represented in the following three forms: (i) confidence distribution (cumulate distribution form), $H_n(\mu) = \Phi(\sqrt{n}(\mu - \bar{x}_n))$; (ii) confidence density (density form), $h_n(\mu) = \frac{1}{\sqrt{2\pi/n}} \exp\{-\frac{n}{2}(\mu - \bar{x}_n)^2\}$; and (iii) confidence curve, $CV_n(\mu) = 2 \min\{\Phi(\sqrt{n}(\mu - \bar{x}_n)), 1 - \Phi(\sqrt{n}(\mu - \bar{x}_n))\}$. See Fig. 29.1 for an illustration. The data are generated from $N(0.3, 1)$ with sample size 100.

Example 29.2.2 ([77]) Suppose the data $X_i \sim N(\mu, \sigma^2)$, $i = 1, \dots, n$, with both unknown μ and σ . A CD for μ is $H_n(\mu) = F_{t_{n-1}}(\frac{\sqrt{n}(\mu - \bar{x}_n)}{s_n})$, where s_n is the sample standard deviation and $F_{t_{n-1}}(\cdot)$ is the cumulative distribution function of student t distribution with parameter $n - 1$. A CD for σ^2 is $H_n(\sigma^2) = 1 - F_{\chi_{n-1}^2}(\frac{(n-1)s_n^2}{\sigma^2})$, where $F_{\chi_{n-1}^2}(\cdot)$ is the cumulative distribution function of the χ_{n-1}^2 -distribution.

Example 29.2.3 ([77]) Let $\hat{\theta}$ be a consistent estimator of θ . For bootstrap, the distribution of $\hat{\theta}^* - \theta$ is estimated by the bootstrap distribution $\hat{\theta}^* - \hat{\theta}$, where $\hat{\theta}^*$ is the estimator of θ computed on a bootstrap sample [26]. An asymptotic CD for θ is given by $H_n(\theta) = 1 - \Pr(\hat{\theta}^* - \hat{\theta} \leq \hat{\theta} - \theta) =$

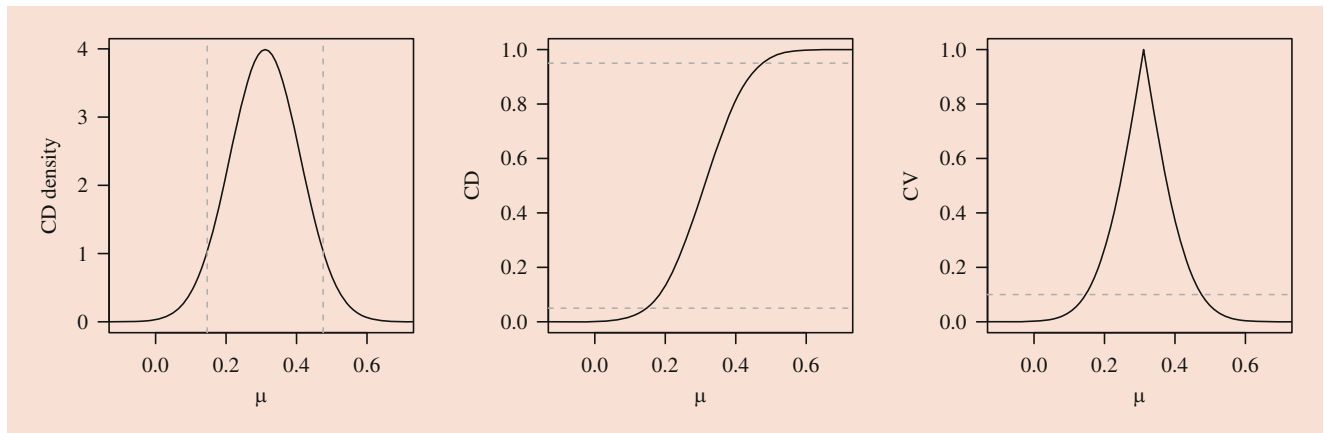


Fig. 29.1 Confidence distribution presented in Example 29.2.1 in the forms of density function, cumulative distribution function, and confidence curve

$\Pr(\hat{\theta}^* \geq 2\hat{\theta} - \theta)$. In addition, when the limiting distribution of normalized $\hat{\theta}$ is symmetric, the raw bootstrap distribution $H_n(\theta) = 1 - \Pr(\hat{\theta} - \hat{\theta}^* \leq \hat{\theta} - \theta) = \Pr(\hat{\theta}^* \leq \theta)$ is also an asymptotic CD.

Example 29.2.4 Suppose we are interested in the location parameter θ of a continuous distribution. When the distribution F is symmetric, i.e., $F(\theta - y) = 1 - F(\theta + y)$, θ is the median. The Wilcoxon rank test for $H_0 : \theta = t, H_1 : \theta \neq t$ is based on the summation of signed ranks of $Y_i - t$, i.e., the test statistic $W = \sum_{i=1}^n Z_i R_i$, where R_i is the rank of $|Y_i - t|$ and Z_i is an indicator variable with 1 if $Y_i - t > 0$ and -1 otherwise. Denote by $p(t)$ the p -value associated with the Wilcoxon rank test for $H_0 : \theta = t, H_1 : \theta \neq t$. When t varies in $(-\infty, \infty)$, the p -value $p(t)$ is referred to as a p -value function. We can prove that the p -value function $p(t)$ is an asymptotic CD [90]. Figure 29.2 provides illustrations of the asymptotic CD density $p'(t)$, the asymptotic CD function $p(t)$, and the asymptotic CV $2 \min\{p(t), 1 - p(t)\}$ for two sample sizes. The data are generated from $N(0, 1)$ with sample sizes $n = 10$ and 100 , respectively.

Example 29.2.5 ([78]) Suppose that there is an independent and identically distributed sample of size n from a semi-parametric model involving multiple parameters. Let $l_n(\theta)$ be the log profile likelihood function and $\mathcal{J}_n(\theta) = -\dot{l}_n(\theta)$ be the observed Fisher information for a scalar parameter of interest θ . Under certain mild assumptions, Theorem 4.1 of [78] proves that, for any given θ ,

$$G_n(\theta) = H_n(\theta) + o_p(1), \text{ where } G_n(\theta) = \frac{\int_{-\infty}^{\theta} \exp\{l_n(x)\} dx}{\int_{-\infty}^{\infty} \exp\{l_n(x)\} dx},$$

$$H_n(\theta) = \Phi\left(\frac{\theta - \hat{\theta}}{\sqrt{\mathcal{J}_n(\hat{\theta})/n}}\right), \hat{\theta} = \arg \max_{\theta} l_n(\theta).$$

Because at the true parameter value $\theta = \theta_0, H_n(\theta_0)$ converges to $U(0, 1)$ as $n \rightarrow \infty$, it follows that $G_n(\theta_0)$ converges to $U(0, 1)$. Thus, $G_n(\theta)$ is an asymptotic CD. From this observation, we see that CD-based inference may subsume a likelihood inference in some occasions.

If the sample X is from a discrete distribution, we can typically invoke a large sample theory to obtain an asymptotic CD to ensure the asymptotic frequentist coverage property, when the sample size is large. However, when the sample size is limited, we sometimes may want to examine the difference between the “distribution estimator” and the $U(0, 1)$ distribution to get a sense of under and over coverage. To expand the concept of CD to cover the cases of discrete distributions with finite sample sizes, we introduce below the notions of lower and upper CDs. The lower and upper CDs provide us inference statements that are associated with under and over coverages at every significant level.

Definition 29.2.2 (Upper and Lower CDs) A function $H_n^+(\cdot) = H_n^+(x, \cdot)$ on $\mathcal{X} \times \Theta \rightarrow [0, 1]$ is said to be an upper CD for a parameter, if (i) for each given $x \in \mathcal{X}$, $H_n(\cdot)$ is a monotonic increasing function on Θ with values ranging within $(0, 1)$ and (ii) at the true parameter value $\theta = \theta_0, H_n^+(\theta_0) \equiv H_n^+(x, \theta_0)$, as a function of the sample x , is stochastically less than or equal to a uniformly distributed random variable $U \sim U(0, 1)$, i.e.,

$$\Pr(H_n^+(X, \theta_0) \leq t) \geq t. \tag{29.1}$$

Correspondingly, a lower CD $H_n^-(\cdot) = H_n^-(x, \cdot)$ for parameter θ can be defined but with (29.1) replaced by $\Pr(H_n^-(X, \theta_0) \leq t) \leq t$ for all $t \in (0, 1)$.

More generally, we also refer to $H_n^+(\cdot)$ and $H_n^-(\cdot)$ as the upper and lower CD, respectively, even when the monotonic

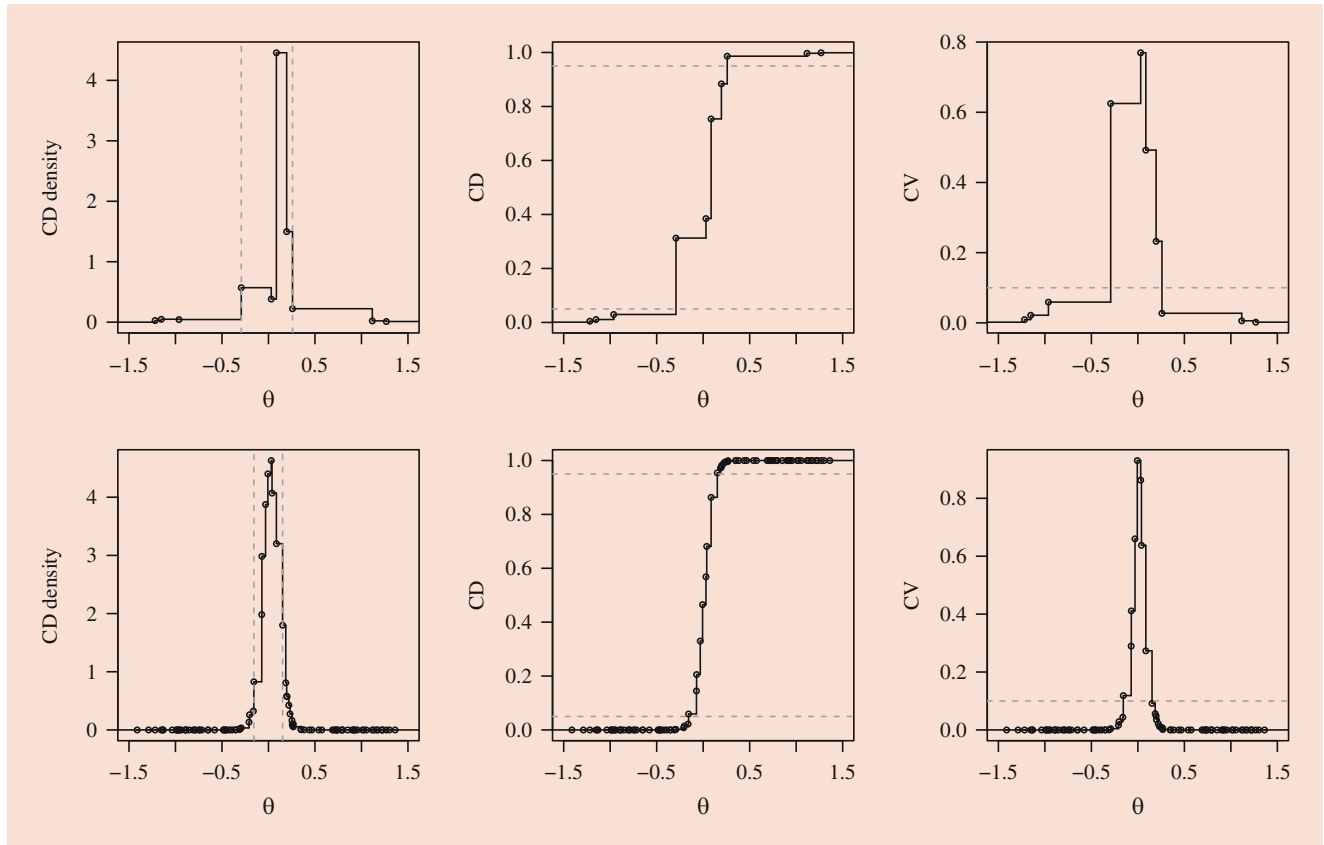


Fig. 29.2 Confidence distributions presented in Example 29.2.4 in the forms of density function, cumulative distribution function, and confidence curve. The top row is for sample size $n = 10$ and the bottom row is $n = 100$

condition (i) is removed. Note that, due to the stochastic dominance inequalities in the definition, we have, for any $\alpha \in (0, 1)$,

$$\Pr(\theta_0 \in \{\theta : H_n^+(X, \theta) \leq \alpha\}) \geq \alpha \text{ and } \Pr(\theta_0 \in \{\theta : H_n^-(X, \theta) \leq \alpha\}) \leq \alpha.$$

Thus, a level- $(1 - \alpha)$ confident interval (or set) $\{\theta : H_n^+(X, \theta) \leq 1 - \alpha\}$ or $\{\theta : H_n^-(X, \theta) \geq \alpha\}$ has guaranteed the coverage rate of $(1 - \alpha)100\%$, regardless of whether we have the monotonic condition in (i). After we remove the monotonic condition in (i), $H_n^+(\cdot)$ and $H_n^-(\cdot)$ may not be a distribution function, and the “nest-ness property” of confidence intervals/sets may also be lost. Here, the “nest-ness property” refers to “a level- $(1 - \alpha)$ confidence set $C_{1-\alpha}$ is not necessarily inside its corresponding level- $(1 - \alpha')$ confidence set $C_{1-\alpha'}$, when $1 - \alpha < 1 - \alpha'$.”

To conclude this section, we present an example of lower and upper CDs.

Example 29.2.6 ([40]) Suppose sample X is from Binomial (n, p_0) with observation x . Let $H_n(p, x) = \Pr(X > x) = \sum_{x < k \leq n} \binom{n}{k} p^k (1 - p)^{n-k}$. We can show that $P(H_n(p_0, X) \leq t) \geq t$ and $P(H_n(p_0, X - 1) \leq t) \leq t$. Thus, $H^+(p, x) =$

$H_n(p, x)$ and $H_n^-(p, x) = H_n(p, x - 1)$ are lower and upper CDs for the success rate p_0 . The half-corrected CD [25,37,72] is

$$\frac{H_n^-(p, x) + H_n^+(p, x)}{2} = \sum_{x < k \leq n_i} \binom{n}{k} p^k (1 - p)^{n-k} + \frac{1}{2} \binom{n}{x} p^x (1 - p)^{n-x}.$$

29.2.2 CD-Based Inference

Analogous to the Bayesian posterior, a CD contains a wealth of information for constructing any type of frequentist inference. We illustrate three aspects of making inference based on a given CD. Figure 29.3 from [90] provides a graphical illustration of the point estimation, confidence interval, and hypothesis testing. More specifically:

Point Estimation The natural choices of point estimators of the parameter θ given a CD $H_n(\cdot)$ include (i) the median $\tilde{\theta}_n = H_n(1/2)$, (ii) the mean $\bar{\theta}_n = \int_{\theta \in \Theta} \theta dH_n(\theta)$, and (iii) the mode $\hat{\theta}_n = \arg \max_{\theta \in \Theta} h_n(\theta)$, where $h_n(\theta) = dH_n(\theta)/d\theta$

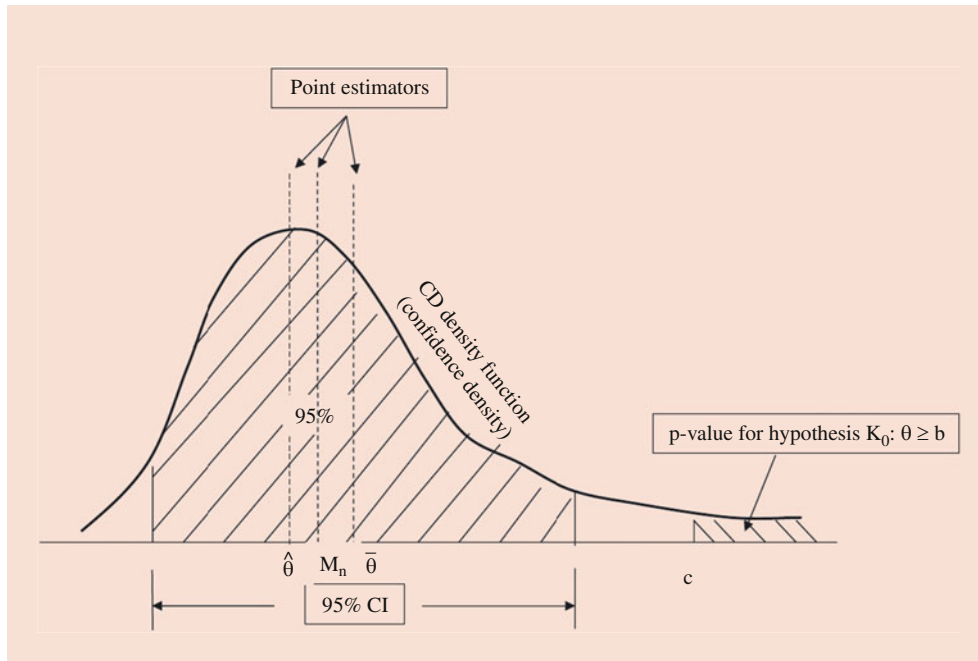


Fig. 29.3 A graphical illustration of CD-based inference [90]

is the confidence density function. Under some moderate conditions, these three point estimators are consistent [77, 90, 91].

To further understand these three types of estimators, the median $\tilde{\theta}_n$ is an unbiased estimator with $\Pr_{\theta_0}(\tilde{\theta}_n \leq \theta_0) = \Pr_{\theta_0}(1/2 \leq H_n(\theta_0)) = 1/2$. The mean $\bar{\theta}_n$ can be viewed as a frequentist analog of Bayesian estimator under the squared loss function. The mode $\hat{\theta}_n$ matches with the maximum likelihood estimator if the confidence density is from a normalized likelihood function [90].

Confidence Interval As discussed in Sect. 29.2.1, in a confidence curve, a line across the y-axis of the significance level α intersects with the confidence curve at two points, and these two points correspond to an $1 - \alpha$ level, equal-tailed, two-sided confidence interval for θ , i.e., $(H_n^{-1}(\alpha/2), H_n^{-1}(1 - \alpha/2))$. Furthermore, $(-\infty, H_n^{-1}(1 - \alpha)]$ and $[H_n^{-1}(\alpha), \infty)$ are one-sided $1 - \alpha$ level confidence intervals for the parameter θ .

Hypothesis Testing From a CD, one can obtain p -values for various hypothesis testing problems. The natural thinking is to measure the support that $H_n(\cdot)$ lends to a null hypothesis [29]. Xie and Singh [90] summarized making inference for hypothesis testing from a CD in the following theorem.

Theorem 29.2.1 (i) For the one-sided test $K_0 : \theta \in C$ versus $K_1 : \theta \in C^c$, where c denotes the complementary set and C is an interval of the type of $C_l = (-\infty, b]$ or $C_u = [b, \infty)$, we have $\sup_{\theta \in C} \Pr_{\theta}(p(C) \leq \alpha) = \alpha$, and $p(C) = H_n(C)$ is the corresponding p -value of the test. (ii)

For the singleton test $K_0 : \theta = b$ versus $K_1 : \theta \neq b$, we have $\Pr_{\theta=b}(2 \min\{p(C_l), p(C_u)\} \leq \alpha) = \alpha$, and $2 \min\{p(C_l), p(C_u)\} = 2 \min\{H_n(b), 1 - H_n(b)\}$ is the p -value of the corresponding test.

Example 29.2.7 ([90]) Consider Example 29.2.2 again. A CD for θ is $H_n = F_{t_{n-1}}(\frac{\sqrt{n}(\mu - \bar{x}_n)}{s_n})$. For a one-sided test $K_0 : \mu \leq b$ versus $K_1 : \mu > b$, its support on the null set $C = (-\infty, b]$ is

$$p(C) = p((-\infty, b]) = H_n(b) = F_{t_{n-1}}(\frac{\sqrt{n}(b - \bar{x}_n)}{s_n}).$$

This is the same p -value using the one-sided t-test. For a two-sided test $K_0 : \theta = b$ versus $K_1 : \theta \neq b$, the null set $C = \{b\}$. We would like to measure the supports of two alternative sets $p(C_l^c)$ and $p(C_u^c)$. The rejection region is defined as $\{x : 2 \max\{p(C_l^c), p(C_u^c)\} \geq 1 - \alpha\}$, i.e.,

$$\{x : 2 \min\{p(C_l), p(C_u)\} \leq \alpha\} = \{x : 2 \min\{H_n(b), 1 - H_n(b)\} \leq \alpha\}. \tag{29.2}$$

Under K_0 with $\theta = b$, $2 \min\{p(C_l), p(C_u)\} = 2 \min\{H_n(b), 1 - H_n(b)\} \sim U(0, 1)$ by the definition of a CD. Thus,

$$\Pr_{\theta=b}(2 \min\{p(C_l), p(C_u)\} \leq \alpha) = \Pr_{\theta=b}(2 \min\{H_n(b), 1 - H_n(b)\} \leq \alpha) = \alpha$$

and the reject region (29.2) corresponds to a level α test. Again, the p -value $2 \min\{p(C_l), p(C_u)\}$ is the standard p -value from a two-sided t-test.

29.2.3 Combination of CDs for Fusion Learning

One of the important applications of CD development is on fusion learning, which synthesizes information from disparate sources with deep implications for meta-analysis [12, 15, 40, 51–53, 72, 75, 77, 81, 91, 92]. Fusion learning aims to combine inference results obtained from different data sources to achieve a more efficient overall inference result. CD-based fusion learning applies even when inference results are derived from different tests or different paradigms, i.e., Bayesian, fiducial, and frequentist (BFF).

The combination of CD can be considered as a unified framework for fusion learning. Suppose there are k independent studies that are dedicated to estimate a common parameter of interest θ . We assume that we have a CD $H^i(\cdot)$ for θ for the sample x_i of the i -th study. Singh et al. [77] proposed a general recipe for combining these k independent CDs:

$$H^c(\theta) \equiv G_c\{g_c(H^1(\theta), \dots, H^k(\theta))\}, \quad (29.3)$$

where g_c is a given continuous function on $[0, 1]$ which is nondecreasing in each coordinate, the function G_c is determined by the monotonic function g_c with $G_c(t) = \Pr(g_c(U_1, \dots, U_k) \leq t)$, and U_1, \dots, U_k are independent uniform random variables. The function $H_c(\cdot)$ contains information from all k samples and is referred to as a combined CD for the parameter θ . Furthermore, the CD obtained by Eq. (29.3) does not require any information regarding how the input CDs are obtained.

A special class of the general combining framework (29.3) plays a prominent role in unifying many modern meta-analysis approaches. The choice of the function g_c for this special class is

$$g_c(u_1, \dots, u_k) = w_1 F^{-1}(u_1) + \dots + w_k F^{-1}(u_k), \quad (29.4)$$

where $F(\cdot)$ is a given cumulative distribution function and $w_i \geq 0$ with at least one $w_i \neq 0$ are generic weights for the combination rule. Generally, there are two types of weights: fixed weights to improve the efficiency of combination and adaptive weights based on data.

As shown in [91], it is remarkable that by choosing different g_c functions, all the classic approaches of combining p -values including Fisher, Normal (Stouffer), Min (Tippett), Max, and Sum methods [55] and all the five model-based meta-analysis estimators described in [67] including the maximum likelihood method and Bayesian approach under fixed-effects model, method of moment estimators, restricted maximum likelihood method, and Bayesian estimator with a normal prior under random-effects model, can all be obtained through a CD combination framework. Furthermore, it was shown in [94] that Mantel-Haenszel and Peto methods as well

as Tian et al.'s method of combining confidence intervals [81] for meta-analysis of 2×2 tables can also all be obtained through a CD combination framework. An R-package "gmeta" developed by [95] implements the CD combining framework for fusion learning including classical p -value combination methods from [55], meta-analysis estimators with both fixed-effects and random-effects models, and many other approaches.

Fusion learning under the framework of combining CD provides an extensive and powerful tool for synthesizing information from diverse data sources. This approach has sound theoretical support and has been applied to many practical situations including robust fusion learning [91], exact fusion learning for discrete data [52, 81], fusion learning for heterogeneous studies [53], nonparametric fusion learning [15, 51], split-conquer-combine approach [12], individualized fusion learning (i -fusion) [75], etc. We refer to [13] for more detailed discussions.

29.2.4 Multivariate CDs

A simultaneous CD for vector parameters can sometimes be difficult to define [72], especially on how to define a multivariate CD in the exact sense in some non-Gaussian settings to ensure that their marginal distributions are CDs for the corresponding single parameter. We consider the Behrens-Fisher problem of testing for the equality of means from two multivariate normal distributions when the covariance matrices are unknown and possibly not equal. A joint CD of the two population means (μ_1, μ_2) has a joint density of the form

$$f_1 \left(\frac{\mu_1 - \bar{x}_1}{s_1/\sqrt{n_1}} \right) f_2 \left(\frac{\mu_2 - \bar{x}_2}{s_2/\sqrt{n_2}} \right) / (s_1 s_2 \sqrt{n_1 n_2}),$$

where f_i is the density function for the student t -distribution with $n_i - 1$ degrees of freedom, $i = 1, 2$. The marginal distribution of $\mu_1 - \mu_2$ is only an asymptotic CD but not a CD in the exact sense.

The good news in the multidimensional case is that under asymptotic settings or wherever bootstrap theory applies, one can still work with multivariate CDs [90]. When no analytic confidence curve for the parameter vector θ of interest is available, the product method of [4] can be used if confidence curves are available for each component of the vector [72]. Additionally, if we only consider center-outward confidence regions instead of all Borel sets in the $p \times 1$ parameter space, the central-CDs considered in [78] and the confidence net considered in [71] offer coherent notions of multivariate CDs in the exact sense [90].

There are many approaches to obtain CDs. One way is normalizing a likelihood function curve with respect to its

parameters so that the area underneath the curve is one. The normalized likelihood function is typically a density function. For instance, under some mild conditions, Fraser and McDunnough [32] show that this normalized likelihood function is the normal density function of an asymptotic CD. Other ways like bootstrap distributions and p -value functions also often provide valid CDs. Finally, CDs and fiducial distributions have been always linked since their inception. The class of fiducial inference provides another systematic way to obtain CDs and we will further discuss fiducial inference in the next section.

29.3 Fiducial Inference

CD can be somehow viewed as “the Neymanian interpretation of Fisher’s fiducial distributions” [74]. From the definition of CD and fiducial distribution, we may consider the fiducial distribution as one special type of CD, though the CD looks at the problem of obtaining an inferentially meaningful distribution on the parameter space from a pure frequentist point of view [90]. Nevertheless, fiducial inference provides a systematic way to obtain a CD, and its development provides a rich class of literature for CD inference. We briefly review fiducial inference and its recent developments in this section.

29.3.1 Fiducial Inference

R.A. Fisher introduced the idea of fiducial probability and fiducial inference [28] as a potential replacement of the Bayesian posterior distribution. Although he discussed fiducial inference in several subsequent papers, there appears to be no rigorous definition of a fiducial distribution for a vector parameter. The basic idea of the fiducial argument is switching the role of data and parameters to introduce the distribution on the parameter space. This obtained distribution then summarizes our knowledge about the unknown parameter. Since the mid-2000s, there has been a renewed interest in modern modifications of fiducial inference. The common approaches for these modifications rely on a definition of inferentially meaningful probability statements about subsets of the parameter space without introducing any prior information.

These modern approaches include generalized fiducial inference [37, 41], Dempster-Shafer theory [22, 24], and inferential models [56, 61]. Objective Bayesian inference, which aims at finding nonsubjective model-based priors, can also be seen as addressing the same question. Examples of recent breakthroughs related to reference prior and model selection are [2, 7, 8]. Another related approach is based on higher-order likelihood expansions and implied data-dependent

priors [30, 31, 33–36]. There are many more references that interested readers can find in [41].

29.3.2 Generalized Fiducial Distribution

Generalized fiducial inference, motivated by [83, 84], has been at the forefront of the modern fiducial revival. Generalized fiducial inference defines a data-dependent measure on the parameter space by using an inverse of a deterministic data generating equation without the use of Bayes theorem.

Motivated by Fisher’s fiducial argument, generalized fiducial inference begins with expressing the relationship between the data Y and the parameters θ as

$$Y = G(U, \theta), \quad (29.5)$$

where $G(\cdot, \cdot)$ is a deterministic function termed as the data generating equation and U is the random component of this data generating equation whose distribution is independent of parameters and completely known.

The data Y are created by generating a random variable U and plugging it into the data generating equation (29.5). For example, a single observation from $N(\mu, 1)$ distribution can be written as $Y = \mu + U$, where $\theta = \mu$ and U is $N(0, 1)$ random variable.

Fisher’s original fiducial argument only addresses the simple case where the data generating equation (29.5) can be inverted and the inverse $Q_y(u) = \theta$ exists for any observed y and for any arbitrary u . One can define the fiducial distribution for θ as the distribution of $Q_y(U^*)$ where U^* is an independent copy of U . Equivalently, a sample from the fiducial distribution of θ can be obtained by first generating U_i^* , and then let $\theta_i^* = Q_y(U_i^*)$, $i = 1, \dots, n$. Point estimation and confidence intervals for θ can be obtained based on this sample. In the $N(\mu, 1)$ example, $Q_y(u) = y - u$ and the fiducial distribution is therefore the distribution of $y - U^* \sim N(y, 1)$.

In the case of no θ satisfying Eq. (29.5), Hannig [37] proposed to use the distribution of U conditional on the event $\{u : y = G(u, \theta), \text{ for some } \theta\}$. Hannig et al. [41] generalized this approach and proposed an attractive definition of generalized fiducial distribution (GFD) through a weak limit.

Definition 29.3.1 A probability measure on the parameter space Θ is called a GFD if it can be obtained as a weak limit

$$\lim_{\epsilon \rightarrow 0} \left[\arg \min_{\theta^*} \|y - G(U^*, \theta^*)\| \mid \min_{\theta^*} \|y - G(U^*, \theta^*)\| \leq \epsilon \right]. \quad (29.6)$$

Hannig et al. [41] pointed out a close relationship between GFD and approximate Bayesian computations (ABC) [3].

In an idealized ABC, one first generates an observation θ^* from the prior, then generates a new sample using a data generating equation $y^* = G(U^*, \theta^*)$, and compares the generated data with the observed data y . If the observed and generated datasets are close, i.e., $\|y - y^*\| \leq \epsilon$, the generated θ^* is accepted; otherwise it is rejected and the procedure is repeated. On the other hand, as for GFD, one first generates U^* , finds a best fitting $\theta^* = \arg \min_{\theta^*} \|y - G(U^*, \theta^*)\|$, computes $y^* = G(U^*, \theta^*)$, again accepts θ^* if $\|y - y^*\| \leq \epsilon$, and rejects otherwise. In either approach an artificial dataset $y^* = G(U^*, \theta^*)$ is generated and compared to the observed data. The main difference is that the Bayes posterior simulates the parameter θ^* from the prior, while GFD uses the best-fitting parameter.

Fiducial distributions often have good frequentist properties, and corresponding fiducial confidence intervals often give asymptotically correct coverage [37, 41]. In addition, fiducial distribution is a data-dependent measure on the parameter space and thereby a CD. Xie and Singh [90] described the relation between the concepts of CD and fiducial distributions using an analogy in point estimation: A CD is analogous to a consistent estimator and a fiducial distribution is analogous to a maximum likelihood estimator. In the context of point estimation, a consistent estimator does not have to be a maximum likelihood estimator. But under some regularity conditions, the maximum likelihood estimator typically provides a standard procedure to obtain a consistent estimator. In the context of distribution estimator, a CD does not have to be a fiducial distribution. However, under suitable conditions, a fiducial distribution often has good frequentist properties and thus a CD.

29.3.3 A User-Friendly Formula for GFD

While Definition (29.6) for GFD is conceptually and mathematically appealing, it is not clear how to compute the limit in most of practical situations. The following theorem proposed by [41] provides a computational tool.

Theorem 29.3.1 *Under certain assumptions, the limiting distribution in (29.6) has a density*

$$r(\theta|y) = \frac{f(y, \theta)J(y, \theta)}{\int_{\Theta} f(y, \theta')J(y, \theta') d\theta'}, \tag{29.7}$$

where $f(y, \theta)$ is the likelihood and the function

$$J(y, \theta) = D \left(\frac{d}{d\theta} G(u, \theta) \Big|_{u=G^{-1}(y, \theta)} \right). \tag{29.8}$$

If (i) $n = p$, then $D(A) = |\det A|$. Otherwise the function $D(A)$ depends on the norm used; (ii) the l_∞

norm gives $D(A) = \sum_{i=(i_1, \dots, i_p)} |\det(A)_i|$;¹ (iii) under an additional assumption stated in [41], the l_2 norm gives $D(A) = (\det A^\top A)^{1/2}$.

Hannig et al. [41] recommended using (ii) for practitioners. A nice property of GFD is that GFD is invariant under smooth re-parameterizations. This property follows directly from (29.6), since for an appropriate selection of minimizers and any one-to-one function $\theta = \phi(\eta)$,

$$\phi \left(\arg \min_{\eta^*} \|y - G(U^*, \phi(\eta^*))\| \right) = \arg \min_{\theta^*} \|y - G(U^*, \theta^*)\|.$$

Note that GFD could change with transformations of the data generating equation. Assume that the observed dataset has been transformed by a one-to-one smooth transformation $Z = T(Y)$. By the chain rule, the GFD based on this new data generating equation and observed data $z = T(y)$ is the density (29.7) with the Jacobian function

$$J_T(z, \theta) = D \left(\frac{d}{dy} T(y) \cdot \frac{d}{d\theta} G(u, \theta) \Big|_{u=G^{-1}(y, \theta)} \right), \tag{29.9}$$

where for simplicity we write y instead of $T^{-1}(z)$.

29.3.4 Examples of GFD

In this section we will consider two examples, linear regression and uniform distribution. In the first case, the GFD is the same as Bayes posterior with respect to the independence Jeffreys prior, while in the second case, the GFD is not a Bayes posterior with respect to any prior (that is not data dependent).

Linear Regression [41] We consider a generalized fiducial approach to regression problem. We express linear regression via the data generating equation,

$$Y = G(U, \theta) = X\beta + \sigma U,$$

where Y is the dependent variables, X is the design matrix, $\theta = (\beta, \sigma)$ are the unknown parameters, and U is a random vector with known density $f(u)$ independent of θ and X . Note that $\frac{d}{d\theta} G(U, \theta) = (X, U)$ and $U = (y - X\beta)/\sigma$; the Jacobian in (29.9) using the l_∞ norm simplifies to

¹In (ii) the sum spans over $\binom{n}{p}$ of p -tuples of indexes $i = (1 \leq i_1 < \dots < i_p \leq n)$. For any $n \times p$ matrix A , the sub-matrix $(A)_i$ is the $p \times p$ matrix containing the rows $i = (i_1, \dots, i_p)$ of A .

$$J_\infty(y, \theta) = \sigma^{-1} \sum_{\substack{i=(1, \dots, i_p) \\ 1 \leq i_1 < \dots < i_p \leq n}} |\det(X, Y)_i|,$$

and the density of GFD is

$$r(\beta, \sigma | y) \propto \sigma^{-n-1} f((Y - X\beta)/\sigma).$$

The fiducial solution is the same as the Bayesian solution using Jeffreys prior [5]. Furthermore, by a simple calculation, the Jacobian with l_2 norm differs from $J_\infty(y, \theta)$ only by a constant; the GFD remains unchanged.

GFD in Irregular Models [41] We consider an irregular model $U(a(\theta) - b(\theta), a(\theta) + b(\theta))$. The reference prior for

this model has been shown complex in Theorem 8 from [7]. Considering GFD approach, we first express the observed data by the following data generating equation:

$$Y_i = a(\theta) + b(\theta)U_i, \quad U_i \stackrel{i.i.d.}{\sim} U(-1, 1).$$

By simple algebra,

$$\frac{d}{d\theta} G(u, \theta) = a'(\theta) + b'(\theta)U \text{ with } U = b^{-1}(\theta)(Y - a(\theta)).$$

If $a'(\theta) > |b'(\theta)|$, (29.8) simplifies to

$$J_1(y, \theta) = n[a'(\theta) - a(\theta)\{\log b(\theta)\}' + \bar{y}_n\{\log b(\theta)\}'],$$

and the GFD is

$$r_1(\theta|y) \propto \frac{a'(\theta) - a(\theta)\{\log b(\theta)\}' + \bar{y}_n\{\log b(\theta)\}'}{b(\theta)^n} I_{\{a(\theta)-b(\theta) < y_{(1)} \ \& \ a(\theta)+b(\theta) > y_{(n)}\}}.$$

Consider an alternative fiducial solution, which constructs the GFD based on the minimal sufficient and ancillary statistics $Z = \{h_1(Y_{(1)}), h_2(Y_{(n)}), (Y - Y_{(1)})/(Y_{(n)} - Y_{(1)})\}^T$, where

$Y_{(1)}, Y_{(n)}$ are order statistics, $h_1^{-1}(\theta) = EY_{(1)} = a(\theta) - b(\theta)(n - 1)/(n + 1)$ and $h_2^{-1}(\theta) = EY_{(n)} = a(\theta) + b(\theta)(n - 1)/(n + 1)$. By a simple calculation,

$$J_2(y, \theta) = (w_1 + w_2) \left[a'(\theta) - a(\theta)\{\log b(\theta)\}' + \frac{w_1 y_{(1)} + w_2 y_{(n)}}{w_1 + w_2} \{\log b(\theta)\}' \right],$$

$$r_2(\theta|y) \propto \frac{I_{\{a(\theta)-b(\theta) < y_{(1)} \ \& \ a(\theta)+b(\theta) > y_{(n)}\}}}{[(w_1 + w_2)[a'(\theta) - a(\theta)\{\log b(\theta)\}'] + (w_1 y_{(1)} + w_2 y_{(n)})\{\log b(\theta)\}']^{-1} b(\theta)^n},$$

where $w_1 = h_1'(y_{(1)})$ and $w_2 = h_2'(y_{(n)})$.

Hannig et al. [41] performed extensive simulation studies for a particular case $U(\theta, \theta^2)$ comparing GFD to the Bayesian posteriors with the reference prior $\pi(\theta) = \frac{(2\theta-1)}{\theta(\theta-1)} e^{\psi(\frac{2\theta}{\theta-1})}$ [7]² and flat prior $\pi(\theta) = 1$. The simple GFD, the alternative GFD, and the reference prior Bayes posterior maintain nominal coverage for all parameter settings. However, the flat prior Bayes posterior does not have a satisfactory coverage, with the worst departures from nominal coverage for small sample size and large parameter θ .

Nonparametric Fiducial Inference with Right-Censored Data [17] Let failure times X_i ($i = 1, \dots, n$) follow the true distribution function F_0 and censoring times C_i ($i = 1, \dots, n$) have the distribution function R_0 . We treat the situation when failure and censoring times are independent and unknown.

Suppose we observe right-censored data $\{y_i, \delta_i\}$ ($i = 1, \dots, n$), where $y_i = x_i \wedge c_i$ is the minimum of x_i and c_i , $\delta_i = I\{x_i \leq c_i\}$ denotes censoring indicator.

Consider the following data generating equation:

$$Y_i = F^{-1}(U_i) \wedge R^{-1}(V_i), \quad \Delta_i = I\{F^{-1}(U_i) \leq R^{-1}(V_i)\}$$

$$(i = 1, \dots, n),$$

where U_i, V_i are independent and identically distributed $U(0, 1)$.

For a failure event $\delta_i = 1$, we have full information about failure time x_i , i.e., $x_i = y_i$, and partial information about censoring time c_i , i.e., $c_i \geq y_i$. Thus,

$$F^{-1}(u_i) = y_i \iff F(y_i) \geq u_i, F(y_i - \epsilon) < u_i \text{ for any } \epsilon > 0.$$

For a censored event $\delta_i = 0$, we only know partial information about x_i , i.e., $x_i > y_i$, and full information on c_i , i.e., $c_i = y_i$. Similarly,

² $\psi(x)$ is the digamma function defined by $\psi(z) = \frac{d}{dz} \log(\Gamma(z))$ for $z > 0$, where Γ is gamma function.

$$F^{-1}(u_i) > y_i \iff F(y_i) < u_i,$$

$$R^{-1}(v_i) = y_i \iff R(y_i) \geq v_i, R(y_i - \epsilon) < v_i \text{ for any } \epsilon > 0.$$

The complete inverse map of the data generating equation is

$$Q^F(y, \delta, u) = \left\{ F : \begin{cases} F(y_i) \geq u_i, F(y_i - \epsilon) < u_i \text{ for any } \epsilon > 0 & \text{for all } i \text{ such that } \delta_i = 1 \\ F(y_j) < u_j & \text{for all } j \text{ such that } \delta_j = 0 \end{cases} \right\}, \tag{29.11}$$

$$Q^{F,R}(y, \delta, u, v) = \bigcap_i Q_{\delta_i}^{F,R}(y_i, u_i, v_i) = Q^F(y, \delta, u) \times Q^R(y, \delta, v), \tag{29.10}$$

where

and $Q^R(y, \delta, v)$ is analogous.

Let (U^*, V^*) be an independent copy of (U, V) . Because the inverse (29.10) separates into a Cartesian product, and of the fact that U^* and V^* are independent, the marginal fiducial distribution for the failure distribution function F is

$$Q^F(y, \delta, U^*) \mid \{Q^F(y, \delta, U^*) \neq \emptyset\}.$$

Figure 29.4 from [17] demonstrates the survival function representation of $Q^F(y, \delta, u)$, as defined in Eq. (29.11), for one dataset with $n = 8$ observations of X following $Weibull(20, 10)$ censored by Z following $Exp(20)$. Each of the panels corresponds to a different value of u , where each u is a realization of U^* . Any survival function lying between the upper red and the lower black fiducial survival functions corresponds to an element of the closure of $Q^F(y, \delta, u)$. The technical details of sampling refer to Algorithm 1 in [17]. The corresponding fiducial-based confidence intervals proposed in [17] maintain coverage in situations where asymptotic methods often have substantial coverage problems. Furthermore, as also shown in [17], the average length of their log-interpolation fiducial confidence intervals is often shorter than the length of confidence intervals for competing methods that maintain coverage. As pointed by [80], it would also be interesting to consider other choices of fiducial samples such as monotonic spline interpolation.

GFDs for Discrete Distributions [41] Let Y be a random variable with distribution function $F(y|\theta)$. Assume there is \mathcal{Y} so that $P_\theta(Y \in \mathcal{Y}) = 1$ for all θ , and for each fixed $y \in \mathcal{Y}$, the distribution function is either a nonincreasing function of θ , spanning the whole interval $(0, 1)$, or a constant equal to 1; the left limit $F(y_-|\theta)$ is also either a nonincreasing function of θ spanning the whole interval $(0, 1)$ or a constant equal to 0.

Define $F^-(a|\theta) = \inf\{y : F(y|\theta) \geq a\}$. It is well known [11] that if $U \sim U(0,1)$, $Y = F^-(U|\theta)$ has the correct distribution and we use this association as a data generating equation. It follows that both $Q_y^+(u) = \sup\{\theta : F(y|\theta) = u\}$ and $Q_y^-(u) = \inf\{\theta : F(y_-|\theta) = u\}$ exist and satisfy $F(y|Q_y^+(u)) = u$ and $F(y_-|Q_y^-(u)) = u$. Consequently,

$$P(Q_y^+(u) \leq t) = 1 - F(y|t) \text{ and}$$

$$P(Q_y^-(u) \leq t) = 1 - F(y_-|t).$$

Note that for all $u \in (0, 1)$, the function $F^-(u|\theta)$ is nonincreasing in θ and the closure of the inverse image $\bar{Q}_y(u) = \{Q_y^-(u), Q_y^+(u)\}$. The half-corrected GFD has distribution function

$$R(\theta|y) = 1 - \frac{F(y|\theta) + F(y_-|\theta)}{2}.$$

If either of the distribution functions is constant, we interpret it as a point mass at the appropriate boundary of the parameter space. Analogous argument shows that if the distribution function and its left limit were nondecreasing in θ , the half-corrected GFD would have distribution function

$$R(\theta|y) = \frac{F(y|\theta) + F(y_-|\theta)}{2}.$$

Hannig et al. [41] provide a list of the half-corrected GFDs for three well-known discrete distributions. Let $Beta(0, n + 1)$ and $Beta(x + 1, 0)$ denote the degenerate distributions on 0 and 1, respectively. Let $\Gamma(0, 1)$ denote the degenerate distribution on 0:

- $X \sim \text{Binomial}(m, p)$ with m known. GFD is the mixture of $Beta(x + 1, m - x)$ and $Beta(x, m - x + 1)$ distributions [37].
- $X \sim \text{Poisson}(\lambda)$. GFD is the mixture of $\Gamma(x + 1, 1)$ and $\Gamma(x, 1)$ distributions [22].
- $X \sim \text{Negative Binomial}(r, p)$ with r known. GFD is the mixture of $Beta(r, x - r + 1)$ and $Beta(r, x - r)$ distributions [38].

Model Selection via GFD [41] Hannig and Lee [39] introduced model selection into the generalized fiducial inference paradigm in the context of wavelet regression. Two important ingredients are needed for fiducial model selection: (1) include the choice of model as one of the parameters; (2) include penalization in the data generating equation.

Consider a finite collection of models \mathcal{M} . The data generating equation is

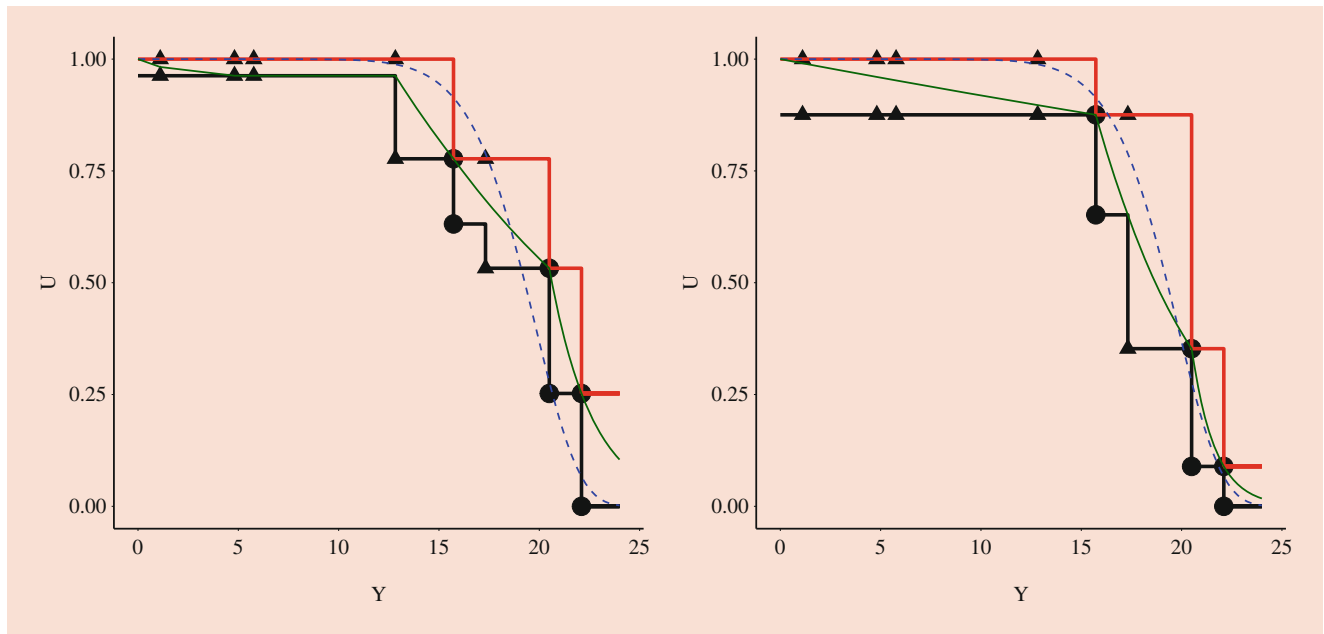


Fig. 29.4 Two realizations of fiducial curves for a sample of size 8 from *Weibull*(20, 10) censored by *Exp*(20) [17]. Here fiducial curves refer to Monte Carlo samples S_i^L , S_i^U , and S_i^I ($i = 1, 2$) from the GFD. The red and black curves are corresponding realizations of the upper

and lower fiducial survival functions. The green curve is the log-linear interpolation type of survival functions. The circle points denote failure observations. The triangle points denote censored observations. The dashed blue curve is the true survival function of *Weibull*(20, 10)

$$Y = G(M, \theta_M, U), \quad M \in \mathcal{M}, \theta_M \in \Theta_M, \quad (29.12)$$

where Y is the observation, M is the model considered, θ_M includes the parameters associated with model M , and U is a random vector of with fully known distribution independent of any parameters. Hannig and Lee [39] proposed a novel way of adding a penalty into the fiducial model selection. In particular, for each model M , they proposed to augment the data generating equation (29.12) by

$$0 = P_k, \quad k = 1, \dots, \min(|M|, n), \quad (29.13)$$

where P_k are independent and identically distributed continuous random variables independent of U with $f_P(0) = q$ and q is a constant determined by the penalty. Hannig and Lee [39] recommended using $q = n^{-1/2}$ as the default penalty. Note that the number of additional equations is the same as the number of unknown parameters in the model. As we never actually observe the outcomes of the extra data generating equations, we will select their values as $p_i = 0$.

For the augmented data generating equation, we have the following theorem from [41]. The quantity $r(M|y)$ can be used for inference in the usual way. For example, fiducial factor, the ratio $r(M_1|y)/r(M_2|y)$, can be used in the same way as a Bayes factor, as discussed in [6] in the context of Bayesian model selection.

Theorem 29.3.2 ([41]) *Suppose $|M| \leq n$ and certain assumptions hold; the marginal generalized fiducial probability of model M is*

$$r(M|y) = \frac{q^{|M|} \int_{\Theta_M} f_M(y, \theta_M) J_M(y, \theta_M) d\theta_M}{\sum_{M' \in \mathcal{M}} q^{|M'|} \int_{\Theta_{M'}} f_{M'}(y, \theta_{M'}) J_{M'}(y, \theta_{M'}) d\theta_{M'}}, \quad (29.14)$$

where $f_M(y, \theta_M)$ is the likelihood and $J_M(y, \theta_M)$ is the Jacobian function computed using (29.9) for each fixed model M .

For more details on the use of fiducial model selection, see [39] and [43].

29.4 Applications and Numerical Examples

29.4.1 CD-Based Inference

Two-Parameter Exponential Distribution Inference procedures based on the two-parameter exponential model, $Exp(\mu, \sigma)$, are extensively used in several areas of statistical practice, including survival and reliability analysis. The probability distribution function and cumulative distribution function of a random variable $X \sim Exp(\mu, \sigma)$ are given, respectively, by

$$f(x) = \frac{1}{\sigma} \exp\left\{-\frac{x-\mu}{\sigma}\right\},$$

$$F(x) = \begin{cases} 1 - \exp\left\{-\frac{x-\mu}{\sigma}\right\} & \text{if } x > \mu, \\ 0 & \text{if } x \leq \mu, \end{cases}$$

and survival function (also known as reliability function) is $S(x) = 1 - F(x)$. The inference problem of interest is to obtain confidence intervals (sets) of μ, σ and $S(t)$ at a given $t > 0$.

Let $X_{(1)}, \dots, X_{(k)}$ be the k ($k > 1$) smallest observations among X_1, \dots, X_n . Then the maximum likelihood estimator of μ and σ are

$$\hat{\mu} = X_{(1)}, \quad \text{and} \quad \hat{\sigma} = \frac{1}{k} \left\{ \sum_{i=1}^k X_{(i)} + (n-k)X_{(k)} - nX_{(1)} \right\}.$$

It turns out that $\hat{\mu}$ and $\hat{\sigma}$ are independent and they follow the distributions

$$U = 2n(\hat{\mu} - \mu)/\sigma \sim \chi^2(2), \quad V = 2k\hat{\sigma}/\sigma \sim \chi^2(2k - 2), \tag{29.15}$$

respectively. Here $\chi^2(m)$ is the chi-square distribution with degree of freedom m . We provide below a simple CD-based method to answer the inference problem of interest.

From Eq. (29.15), we have

$$\frac{n(\hat{\mu} - \mu)}{k\hat{\sigma}} = \frac{U/2}{V/(2k - 2)} \sim F(2, 2k - 2),$$

where $F(a, b)$ is the F -distribution with degrees of freedom a and b . By the pivot-based CD construction method [78, p134], a CD for μ is $H_1(\mu) = 1 - F_{F(2, 2k-2)}(\frac{n(\hat{\mu}-\mu)}{k\hat{\sigma}})$, where $F_{F(2, 2k-2)}$ is the cumulative distribution function of $F(2, 2k - 2)$ -distribution. Similarly, a CD for σ is $H_2(\sigma) = 1 - F_{\chi^2(2k-2)}(\frac{2k\hat{\sigma}}{\sigma})$, where $F_{\chi^2(2k-2)}$ is the cumulative distribution function of $\chi^2(2k - 2)$ -distribution. Inferential statements regarding μ and σ , including confidence intervals and testing results, can be obtained from these two CDs. Coverage rates and test errors obtained from these two CDs are exact.

We can also consider the inference for (μ, σ) jointly. Here, we introduce a simulation-based approach. Let $U^* \sim \chi^2(2)$ and $V^* \sim \chi^2(2k - 2)$ be two independently simulated random numbers. Define

$$\xi^* = \hat{\mu} - \frac{k\hat{\sigma}}{n} \frac{U^*}{V^*} \quad \text{and} \quad \zeta^* = \frac{2k\hat{\sigma}}{V^*}.$$

Then, $\xi^* | (\hat{\mu}, \hat{\sigma}) \sim H_1(\mu)$ and $\zeta^* | (\hat{\mu}, \hat{\sigma}) \sim H_2(\sigma)$, and they are called *CD random variables* [90]. Furthermore, the underlying joint distribution of (ξ^*, ζ^*) , given $(\hat{\mu}, \hat{\sigma})$, is a joint CD function $H_3(\mu, \sigma)$ of (μ, σ) . If we simulate a large number of, say M , copies of (U^*, V^*) , then we can get M copies of (ξ^*, ζ^*) . In order to make inference statements about (μ, σ) , we can treat these M copies of $(\xi_1^*, \zeta_1^*), \dots, (\xi_M^*, \zeta_M^*)$ as if they were M copies of bootstrap estimators in bootstrap inference or as if they were M copies of random samples from the posterior distribution of (μ, σ) in a Bayesian inference.

Additionally, we can also use the M copies of CD random variables $(\xi_1^*, \zeta_1^*), \dots, (\xi_M^*, \zeta_M^*)$ to obtain a pointwise confidence band for $S(t), t > 0$. For each given $t > 0$, we compute $\kappa_j^*(t) = \exp\{-(t - \xi_j^*)/\zeta_j^*\}$, for $j = 1, \dots, M$. Then $[\kappa_{[\alpha M]}^*(t), +\infty)$ and $[\kappa_{[\frac{\alpha}{2} M]}^*(t), \kappa_{[\frac{1-\alpha}{2} M]}^*(t)]$ are the one-sided and two-sided level- α confidence intervals of $S(t)$, respectively, where $\kappa_{[qM]}^*(t)$ is the q -th quantile of $\kappa_1^*(t), \dots, \kappa_M^*(t)$. Now by varying t , $[\kappa_{[\alpha M]}^*(t), +\infty)$ forms a level- α lower confidence band, and $[\kappa_{[\frac{\alpha}{2} M]}^*(t), \kappa_{[\frac{1-\alpha}{2} M]}^*(t)]$ forms a level- α confidence band for the survival function $S(t)$.

We can show that this set of exact confidence bands derived from the CD method matches with those obtained in [69] using Tsui and Weerahandi's generalized inference approach [83], but the CD approach is very simple and more direct. Roy and Mathew [69] illustrated the 95% lower limit $\tilde{S}(t)$ for time ranging from 150 to 2000 in Figure 1 of [69] using a real data example with 19 observations taken from [45]. The data deal with mileages for military personnel carriers that failed in service. Figure 29.5 is a similar plot for the confidence band, using our CD approach with $M = 1000$.

Data [45]:

162, 200, 271, 320, 393, 508, 539, 629, 706, 777, 884, 1008, 1101, 1182, 1463, 1603, 1984, 2355, 2880

Bivariate Normal Correlation Suppose we have the following bivariate normal distribution:

$$N\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}\right),$$

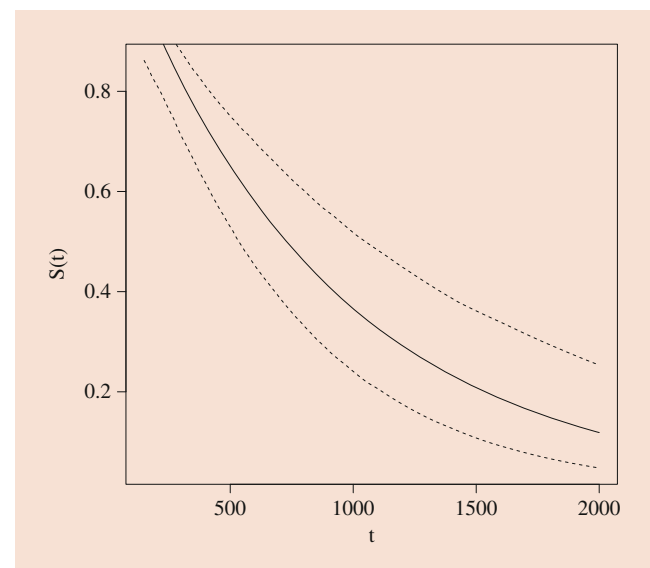


Fig. 29.5 Point estimate (solid line) and 95% confidence band (dashed line) of CD-based inference

and let ρ denote the correlation coefficient. One could use the asymptotic pivot, Fisher's Z [27, 78],

$$\frac{1}{2} \log \frac{1+r}{1-r} - \frac{1}{2} \log \frac{1+\rho}{1-\rho},$$

where r is the sample correlation. The limiting distribution of the above pivot is $N(0, \frac{1}{n-3})$. Therefore, the asymptotic CD is

$$H_n(\rho) = 1 - \Phi \left(\sqrt{n-3} \left[\frac{1}{2} \log \frac{1+r}{1-r} - \frac{1}{2} \log \frac{1+\rho}{1-\rho} \right] \right), \quad -1 \leq \theta \leq 1.$$

Figure 29.6 presents the CD of correlation coefficient ρ for a simulated dataset with $n = 50, \mu_1 = \mu_2 = 1, \sigma_1 = \sigma_2 = 1, \rho = 0.5$.

In addition to the above two examples, there also are recent developments of CDs on causal inference; see more applications in [54].

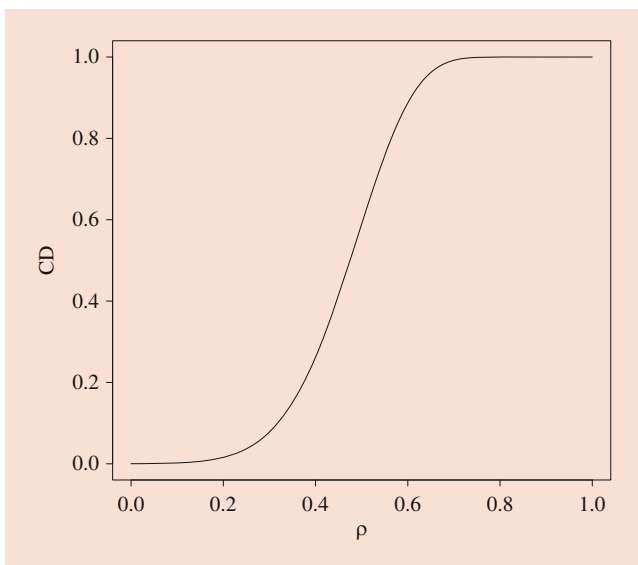


Fig. 29.6 CD of the correlation coefficient ρ

29.4.2 Nonparametric GFD-Based Inference

[17] proposed a fiducial approach to testing reliability function with an infinite dimensional parameter. Their approach does not assume a parametric distribution and is robust to model mis-specification. In [17], they considered a clinical trial of chemotherapy against chemotherapy combined with radiotherapy in the treatment of locally unresectable gastric cancer conducted by the Gastrointestinal Tumor Study Group [70]. In this trial, 45 patients were randomized to each of the 2 groups and followed for several years. The censoring percentage is 13.3% for the combined therapy group, and 4.4% for the chemotherapy group. We are interested in testing whether the two treatment groups have the same survival functions.

The Kaplan-Meier curves for these two datasets are presented in Fig. 29.7a. We notice that the two hazards appear to be crossing, which could pose a problem for some log-rank

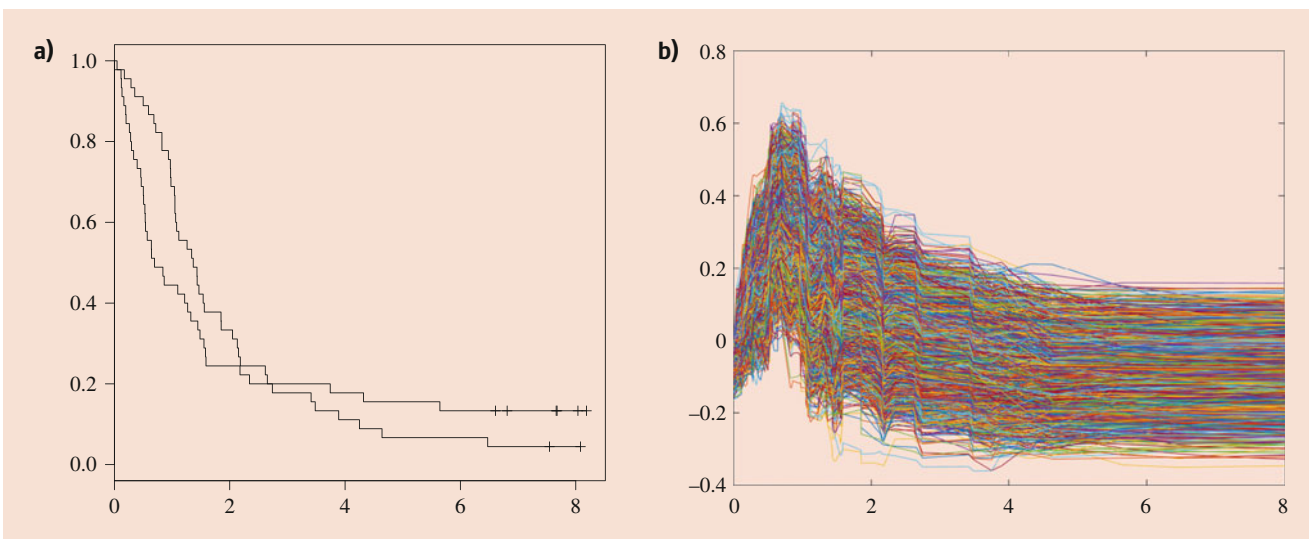


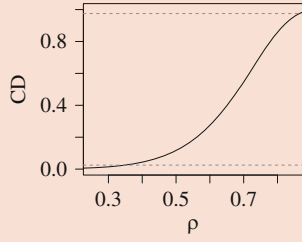
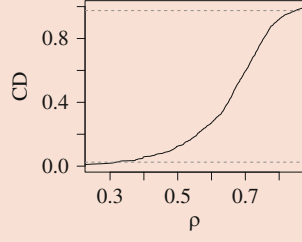
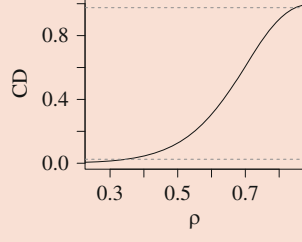
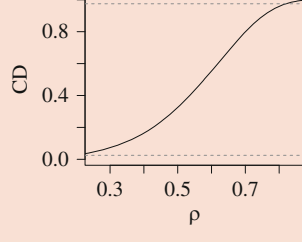
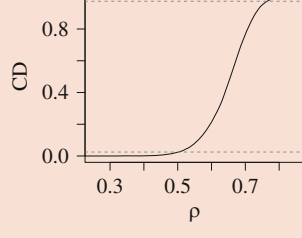
Fig. 29.7 (a) Kaplan-Meier estimators for two treatment groups [17]. (b) Difference of two sample fiducial distributions

tests. In this instance, the fiducial approach gives a small p-value 0.002. The p-values of other types of log-rank tests are reported in [17]. To explain why their proposed fiducial approach works good, they plot the sample of the difference of two fiducial distributions in Fig. 29.7b. If these two datasets are from the same distribution, 0 should be well within the sample curves. However, from Fig. 29.7b, we could see that the majority of curves are very far away from 0 on the interval [0.5, 1]. This gives strong evidence that the group

with combined therapy has significantly worse early survival outcomes.

In [17], they choose to use the sup-norm in the definition of the curvewise confidence intervals and tests. It could be possible to make the procedure more powerful by using a different (possibly weighted) norm [64]. Similarly, it might also be possible to use the choice of norm motivated by inferential models [18, 58, 61]. Besides the above example, there also are recent developments of non-

Table 29.1 Inference on correlation coefficient: combining independent bivariate normal studies

Methods	95% CI	CDs
Fisher's Z method	(0.348,0.845)	
Bootstrap BC_a	(0.317,0.818)	
Profile likelihood	(0.346,0.827)	
Bayes (uniform prior)	(0.188,0.790)	
Combination	(0.505,0.760)	

parametric fiducial inference on interval-censored data and Efron’s empirical Bayes deconvolution; see [19, 20] for more applications.

Data [70]: (* indicates a censored event)

Combination group: 0.05 0.12 0.12 0.13 0.16 0.20 0.20 0.26 0.28 0.30 0.33 0.39 0.46 0.47 0.50 0.51 0.53 0.53 0.54 0.57 0.64 0.64 0.70 0.84 0.86 1.10 1.22 1.27 1.33 1.45 1.48 1.55 1.58 1.59 2.18 2.34 3.74 4.32 5.64 6.61* 6.81* 7.66* 7.68* 8.04* 8.19*

Chemotherapy group: 0.00 0.17 0.29 0.35 0.50 0.59 0.68 0.72 0.82 0.82 0.94 0.97 0.98 0.98 1.04 1.05 1.05 1.06 1.08 1.12 1.26 1.34 1.37 1.43 1.44 1.47 1.54 1.56 1.85 1.85 2.05 2.13 2.15 2.18 2.62 2.65 2.74 3.41 3.48 3.89 4.25 4.64 6.47 7.55* 8.08*

29.4.3 Combining Information from Multiple CDs

We use simple cluster of differentiation 4 (cd-4) count data considered in [23] to demonstrate combining information from CDs. Twenty HIV-positive subjects received an experimental antiviral drug. The cd-4 counts in hundreds were recorded for each subject at baseline and after 1 year of treatment.

We obtained the summary statistics and simulated four independent datasets from the following bivariate normal distribution:

$$N\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}\right),$$

where $\mu_1 = 3.288$, $\mu_2 = 4.093$, $\sigma_1^2 = 0.657$, $\sigma_2^2 = 1.346$, and $\rho = 0.723$.

Suppose each study makes its own inference conclusion individually. Each dataset was analyzed by Fisher’s Z method [27, 76], the bias-corrected and accelerated (BC_a) bootstrap [10, 21, 23], the profile likelihood approach [48], and Bayesian with uniform prior [1], respectively. One natural question we would like to ask is if we can combine the inferences from four independent studies, given that ρ is the same in all studies. The answer is yes. As introduced in Section 29.2.3, combination of CDs is a powerful inferential tool. We fused studies by combining p -values (Stouffer) [55, 95].

The results of the analysis are summarized in Table 29.1. As we can see from the table, four methods in different studies provide more or less similar results, and the combined interval is much shorter than any of the four individual intervals. In order to study the performance of the combination of CDs in this situation, we present a simulation study with 200 replications. Table 29.2 shows the

Table 29.2 Combination of four independent bivariate normal studies via CDs

Methods	Coverage	Mean length (sd) of 95% CIs
Fisher’s Z method	0.948	0.484 (0.140)
Bootstrap BC_a	0.936	0.464 (0.156)
Profile likelihood	0.918	0.436 (0.131)
Bayes (uniform prior)	0.964	0.522 (0.128)
Combination	0.954	0.226 (0.041)

coverage and average length of 95% CIs. We see that not only the combined approach maintains the desired coverage but also the length of CIs is roughly half of the lengths of CIs from individual studies. This result is as expected, since theoretically each study provides a $n^{-1/2}$ -CIs and the sample size of combined data is $4n$ so we expect to obtain $(4n)^{-1/2}$ -CI.

Data [23]:

Baseline: 2.12, 4.35, 3.39, 2.51, 4.04, 5.10, 3.77, 3.35, 4.10, 3.35, 4.15, 3.56, 3.39, 1.88, 2.56, 2.96, 2.49, 3.03, 2.66, 3.00
 One year: 2.47, 4.61, 5.26, 3.02, 6.36, 5.93, 3.93, 4.09, 4.88, 3.81, 4.74, 3.29, 5.55, 2.82, 4.23, 3.23, 2.56, 4.31, 4.37, 2.40

References

1. Baath, R.: Bayesian First Aid. R package (2013)
2. Bayarri, M.J., Berger, J.O., Forte, A., García-Donato, G.: Criteria for Bayesian model choice with application to variable selection. *Ann. Stat.* **40**, 1550–1577 (2012)
3. Beaumont, M.A., Zhang, W., Balding, D.J.: Approximate Bayesian computation in population genetics. *Genetics* **162**, 2025–2035 (2002)
4. Beran, R.: Balanced simultaneous confidence sets. *J. Am. Stat. Assoc.* **83**, 679–686 (1988)
5. Berger, J.: Catalog of Objective Priors. Tech. rep., Duke University (2011)
6. Berger, J.O., Pericchi, L.R.: Objective Bayesian methods for model selection: introduction and comparison. In: *Model Selection*, vol. 38 of IMS Lecture Notes Monogr. Ser., pp. 135–207. Inst. Math. Statist., Beachwood, OH (2001)
7. Berger, J.O., Bernardo, J.M., Sun, D.: The formal definition of reference priors. *Ann. Stat.* **37**, 905–938 (2009)
8. Berger, J.O., Bernardo, J.M., Sun, D.: Objective priors for discrete parameter spaces. *J. Am. Stat. Assoc.* **107**, 636–648 (2012)
9. Birnbaum, A.: Confidence curves: An omnibus technique for estimation and testing statistical hypotheses. *J. Am. Stat. Assoc.* **56**, 246–249 (1961)
10. Cauty, A., Ripley, B.D.: boot: Bootstrap R (S-Plus) Functions. R package version 1.3-23 (2019)
11. Casella, G., Berger, R.L.: *Statistical Inference*, 2nd edn. Wadsworth and Brooks/Cole Advanced Books and Software, Pacific Grove, CA (2002)
12. Chen, X., Xie, M.: A split-and-conquer approach for analysis of extraordinarily large data. *Statistica Sinica* **24**, 1655–1684 (2014)
13. Cheng, J.Q., Liu, R.Y., Xie, M.: Fusion Learning, pp. 1–8 (2017). <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118445112.stat07922>

14. Cisewski, J., Hannig, J.: Generalized fiducial inference for normal linear mixed models. *Ann. Stat.* **40**, 2102–2127 (2012)
15. Claggett, B., Xie, M., Tian, L.: Meta-analysis with fixed, unknown, study-specific parameters. *J. Am. Stat. Assoc.* **109**, 1660–1671 (2014)
16. Cox, D.R.: Some problems connected with statistical inference. *Ann. Math. Stat.* **29**, 357–372 (1958). <https://doi.org/10.1214/aoms/1177706618>.
17. Cui, Y., Hannig, J.: Nonparametric generalized fiducial inference for survival functions under censoring (with discussions and rejoinder). *Biometrika* **106**, 501–518 (2019a). <https://doi.org/10.1093/biomet/asz016>
18. Cui, Y., Hannig, J.: Rejoinder: Nonparametric generalized fiducial inference for survival functions under censoring. *Biometrika* **106**, 527–531 (2019b). <https://doi.org/10.1093/biomet/asz032>
19. Cui, Y., Hannig, J.: A fiducial approach to nonparametric deconvolution problem: discrete case. *Science China Mathematics*. In press
20. Cui, Y., Hannig, J., Kosorok, M.: A unified nonparametric fiducial approach to interval-censored data. (2021) arXiv:2111.14061
21. Davison, A.C., Hinkley, D.V.: *Bootstrap Methods and Their Applications*. Cambridge University Press, Cambridge (1997). <http://statwww.epfl.ch/davison/BMA/>. ISBN 0-521-57391-2
22. Dempster, A.P.: The Dempster-Shafer calculus for statisticians. *Int. J. Approx. Reason.* **48**, 365–377 (2008)
23. DiCiccio, T.J., Efron, B.: Bootstrap confidence intervals. *Stat. Sci.* **11**, 189–228 (1996). <https://doi.org/10.1214/ss/1032280214>
24. Edlefsen, P.T., Liu, C., Dempster, A.P.: Estimating limits from Poisson counting data using Dempster–Shafer analysis. *Ann. Appl. Stat.* **3**, 764–790 (2009)
25. Efron, B.: R.A. Fisher in the 21st century. *Stat. Sci.* **13**, 95–122 (1998)
26. Efron, B., Tibshirani, R.J.: *An Introduction to the Bootstrap*. No. 57 in *Monographs on Statistics and Applied Probability*. Chapman & Hall/CRC, Boca Raton, FL, USA (1993)
27. Fisher, R.A.: Frequency distribution of the values of the correlation coefficient in samples from an indefinitely large population. *Biometrika* **10**, 507–521 (1915). <http://www.jstor.org/stable/2331838>
28. Fisher, R.A.: Inverse probability. *Proc. Camb. Phil. Soc.* **xxvi**, 528–535 (1930)
29. Fraser, D.A.S.: Statistical inference: Likelihood to significance. *J. Am. Stat. Assoc.* **86**, 258–265 (1991)
30. Fraser, D.A.S.: Ancillaries and conditional inference. *Stat. Sci.* **19**, 333–369 (2004)
31. Fraser, D.A.S.: Is Bayes posterior just quick and dirty confidence? *Stat. Sci.* **26**, 299–316 (2011)
32. Fraser, D.A.S., McDunnough, P.: Further remarks on asymptotic normality of likelihood and conditional analyses. *Can. J. Stat. [La Revue Canadienne de Statistique]* **12**, 183–190 (1984). <http://www.jstor.org/stable/3314746>.
33. Fraser, D., Naderi, A.: Exponential models: Approximations for probabilities. *Biometrika* **94**, 1–9 (2008)
34. Fraser, D., Reid, N., Wong, A.: What a model with data says about theta. *Int. J. Stat. Sci.* **3**, 163–178 (2005)
35. Fraser, A.M., Fraser, D.A.S., Staicu, A.-M.: The second order ancillary: A differential view with continuity. *Bernoulli Off. J. Bernoulli Soc. Math. Stat. Probab.* **16**, 1208–1223 (2009)
36. Fraser, D.A.S., Reid, N., Marras, E., Yi, G.Y.: Default Priors for Bayesian and frequentist inference. *J. R. Stat. Soc. Ser. B* **72**, 631–654 (2010)
37. Hannig, J.: On generalized fiducial inference. *Statistica Sinica* **19**, 491–544 (2009)
38. Hannig, J.: Discussion of “On the Birnbaum Argument for the Strong Likelihood Principle” by D. G. Mayo. *Stat. Sci.* **29**, 254–258 (2014)
39. Hannig, J., Lee, T.C.M.: Generalized fiducial inference for wavelet regression. *Biometrika* **96**, 847–860 (2009)
40. Hannig, J., Xie, M.: A note on Dempster-Shafer recombinations of confidence distributions. *Electr. J. Stat.* **6**, 1943–1966 (2012)
41. Hannig, J., Iyer, H., Lai, R.C., Lee, T.C.: Generalized fiducial inference: A review and new results. *J. Am. Stat. Assoc.* **111**, 1346–1361 (2016)
42. Iverson, T.: Generalized fiducial inference. *Wiley Interdiscip. Rev. Comput. Stat.* **6**, 132–143 (2014). <https://onlinelibrary.wiley.com/doi/abs/10.1002/wics.1291>
43. Lai, R.C.S., Hannig, J., Lee, T.C.M.: Generalized fiducial inference for ultra-high dimensional regression. *J. Am. Stat. Assoc.* **110**, 760–772 (2015a)
44. Lai, R.C.S., Hannig, J., Lee, T.C.M.: Generalized fiducial inference for ultrahigh-dimensional regression. *J. Am. Stat. Assoc.* **110**, 760–772 (2015b)
45. Lawless, J.F.: *Statistical Models and Methods for Lifetime Data*. Wiley, New York (1982)
46. Lawless, J.F., Fredette, M.: Frequentist prediction intervals and predictive distributions. *Biometrika* **92**, 529–542 (2005). <https://doi.org/10.1093/biomet/92.3.529>
47. Lehmann, E.L.: The fisher, neyman-pearson theories of testing hypotheses: One theory or two? *J. Am. Stat. Assoc.* **88**, 1242–1249 (1993). <http://www.jstor.org/stable/2291263>.
48. Li, Y., Gillespie, B.W., Shedden, K., Gillespie, J.A.: Profile likelihood estimation of the correlation coefficient in the presence of left, right or interval censoring and missing data. *R J.* **10**, 159–179 (2018). <https://doi.org/10.32614/RJ-2018-040>
49. Liu, Y., Hannig, J.: Generalized fiducial inference for binary logistic item response models. *Psychometrika* **81**, 290–324 (2016)
50. Liu, Y., Hannig, J.: Generalized fiducial inference for logistic graded response models. *Psychometrika* **82**, 1097–1125 (2017)
51. Liu, R.Y., Singh, K.: Notions of limiting p values based on data depth and bootstrap. *J. Am. Stat. Assoc.* **92**, 266–277 (1997)
52. Liu, D., Liu, R.Y., Xie, M.: Exact meta-analysis approach for discrete data and its application to 2×2 tables with rare events. *J. Am. Stat. Assoc.* **109**, 1450–1465 (2014)
53. Liu, D., Liu, R.Y., Xie, M.: Multivariate meta-analysis of heterogeneous studies using only summary statistics: Efficiency and robustness. *J. Am. Stat. Assoc.* **110**(509), 326–340 (2015)
54. Luo, X., Dasgupta, T., Xie, M., Liu, R.: Leveraging the fisher randomization test using confidence distributions: inference, combination and fusion learning. Preprint (2020). arXiv:2004.08472
55. Marden, J.I.: Sensitive and sturdy p-values. *Ann. Stat.* **19**, 918–934 (1991). <http://www.jstor.org/stable/2242091>
56. Martin, R.: *Inferential Models*, pp. 1–8. American Cancer Society (2017). <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118445112.stat07997>
57. Martin, R.: On an inferential model construction using generalized associations. *J. Stat. Plann. Infer.* **195**, 105–115 (2018). <http://www.sciencedirect.com/science/article/pii/S0378375816301537>. Confidence distributions
58. Martin, R.: Discussion of Nonparametric generalized fiducial inference for survival functions under censoring. *Biometrika* **106**, 519–522 (2019). <https://doi.org/10.1093/biomet/asz022>
59. Martin, R., Liu, C.: Inferential models: A framework for prior-free posterior probabilistic inference. *J. Am. Stat. Assoc.* **108**, 301–313 (2013)
60. Martin, R., Liu, C.: Conditional inferential models: combining information for prior-free probabilistic inference. *J. R. Stat. Soc., Ser. B* **77**, 195–217 (2015a)

61. Martin, R., Liu, C.: Inferential Models: Reasoning with Uncertainty. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. CRC Press (2015b). <https://books.google.com/books?id=OdSYCgAAQBAJ>
62. Martin, R., Liu, C.: Marginal inferential models: prior-free probabilistic inference on interest parameters. *J. Am. Stat. Assoc.* **110**, 1621–1631 (2015c)
63. Martin, R., Zhang, J., Liu, C.: Dempster-Shafer theory and statistical inference with weak beliefs. *Stat. Sci.* **25**, 72–87 (2010)
64. Nair, V.N.: Confidence bands for survival functions with censored data: a comparative study. *Technometrics* **26**, 265–275 (1984)
65. Neupert, S.D., Hannig, J.: BFF: Bayesian, fiducial, frequentist analysis of age effects in daily diary data. *J. Gerontol. Ser. B* (2019). <https://doi.org/10.1093/geronb/gbz100>. Gbz100
66. Neyman, J.: Fiducial argument and the theory of confidence intervals. *Biometrika* **32**, 128–150 (1941). <http://www.jstor.org/stable/2332207>
67. Normand, S.-L.T.: Meta-analysis: formulating, evaluating, combining, and reporting. *Stat. Med.* **18**, 321–359 (1999)
68. Qiu, Y., Zhang, L., Liu, C.: Exact and efficient inference for partial bayes problems. *Electron. J. Stat.* **12**, 4640–4668 (2018). <https://doi.org/10.1214/18-EJS1511>
69. Roy, A., Mathew, T.: A generalized confidence limit for the reliability function of a two-parameter exponential distribution. *J. Stat. Plann. Infer.* **128**, 509–517 (2005)
70. Schein, P.S.: A comparison of combination chemotherapy and combined modality therapy for locally advanced gastric carcinoma. *Cancer* **49**, 1771–1777 (1982)
71. Schweder, T.: Confidence nets for curves. *Adv. Stat. Model. Infer. Essays Honor Kjell A. Doksum*, 593–609 (2007)
72. Schweder, T., Hjort, N.L.: Confidence and likelihood. *Scand. J. Stat.* **29**, 309–332 (2002)
73. Schweder, T., Hjort, N.L.: Frequentist analogues of priors and posteriors. *Econ. Philos. Econ.*, 285–317 (2003)
74. Schweder, T., Hjort, N.L.: Confidence, Likelihood, Probability, vol. 41. Cambridge University Press (2016)
75. Shen, J., Liu, R.Y., Xie, M.: ifusion: Individualized fusion learning. *J. Am. Stat. Assoc.* **0**, 1–17 (2019)
76. Signorell, A., et al.: DescTools: Tools for Descriptive Statistics. <https://cran.r-project.org/package=DescTools>. R package version 0.99.28 (2019)
77. Singh, K., Xie, M., Strawderman, W.E.: Combining information from independent sources through confidence distributions. *Ann. Stat.* **33**, 159–183 (2005)
78. Singh, K., Xie, M., Strawderman, W.E.: Confidence distribution (cd): Distribution estimator of a parameter. *Lect. Notes Monogr. Ser.* **54**, 132–150 (2007). <http://www.jstor.org/stable/20461464>
79. Taraldsen, G., Lindqvist, B.H.: Fiducial theory and optimal inference. *Ann. Stat.* **41**, 323–341 (2013)
80. Taraldsen, G., Lindqvist, B.H.: Discussion of Nonparametric generalized fiducial inference for survival functions under censoring. *Biometrika* **106**, 523–526 (2019). <https://doi.org/10.1093/biomet/asz027>
81. Tian, L., Cai, T., Pfeffer, M.A., Piankov, N., Cremieux, P.-Y., Wei, L.: Exact and efficient inference procedure for meta-analysis and its application to the analysis of independent 2×2 tables with all available data but without artificial continuity correction. *Biostatistics* **10**, 275–281 (2008)
82. Tian, L., Wang, R., Cai, T., Wei, L.-J.: The highest confidence density region and its usage for joint inferences about constrained parameters. *Biometrics* **67**, 604–10 (2011)
83. Tsui, K.-W., Weerahandi, S.: Generalized p -values in significance testing of hypotheses in the presence of nuisance parameters. *J. Am. Stat. Assoc.* **84**, 602–607 (1989)
84. Tsui, K.-W., Weerahandi, S.: Corrections: Generalized p -values in significance testing of hypotheses in the presence of nuisance parameters. *J. Am. Stat. Assoc.* **84**(406), 602–607 (1991); MR1010352 (90g:62047); *J. Am. Stat. Assoc.* **86**, 256 (1989)
85. Veronese, P., Melilli, E.: Fiducial and Confidence Distributions for Real Exponential Families. *Scand. J. Stat.* (2014, in press)
86. Wandler, D.V., Hannig, J.: Generalized fiducial confidence intervals for extremes. *Extremes* **15**, 67–87 (2012)
87. Wang, Y.H.: Fiducial intervals: what are they? *Am. Stat.* **54**, 105–111 (2000)
88. Williams, J.P., Hannig, J.: Nonpenalized variable selection in high-dimensional linear model settings via generalized fiducial inference. *Ann. Stat.* **47**, 1723–1753 (2019). <https://doi.org/10.1214/18-AOS1733>
89. Williams, J.P., Storlie, C.B., Therneau, T.M., Jr., C.R.J., Hannig, J.: A bayesian approach to multistate hidden markov models: Application to dementia progression. *J. Am. Stat. Assoc.* **0**, 1–21 (2019)
90. Xie, M., Singh, K.: Confidence distribution, the frequentist distribution estimator of a parameter: A review. *Int. Stat. Rev.* **81**, 3–39 (2013)
91. Xie, M., Singh, K., Strawderman, W.E.: Confidence distributions and a unified framework for meta-analysis. *J. Am. Stat. Assoc.* **106**, 320–333 (2011)
92. Xie, M., Liu, R.Y., Damaraju, C.V., Olson, W.H.: Incorporating external information in analyses of clinical trials with binary outcomes. *Ann. Appl. Stat.* **7**, 342–368 (2013)
93. Xu, X., Li, G.: Fiducial inference in the pivotal family of distributions. *Sci. China Ser. A Math.* **49**, 410–432 (2006)
94. Yang, G., Liu, D., Wang, J., Xie, M.: Meta-analysis framework for exact inferences with application to the analysis of rare events. *Biometrics* **72**, 1378–1386 (2016). <https://onlinelibrary.wiley.com/doi/abs/10.1111/biom.12497>
95. Yang, G., Cheng, J.Q., Xie, M., Qian, W.: gmeta: Meta-Analysis via a Unified Framework of Confidence Distribution. <https://CRAN.R-project.org/package=gmeta>. R package version 2.3-0 (2017)
96. Zhang, J., Liu, C.: Dempster-Shafer inference with weak beliefs. *Statistica Sinica* **21**, 475–494 (2011)



Yifan Cui received his Ph.D. from University of North Carolina at Chapel Hill in 2018. He is currently a tenure track faculty member at the Center for Data Science in Zhejiang University (ZJU). He has worked at Wharton Statistics Department, University of Pennsylvania as a postdoctoral researcher and Department of Statistics and Data Science, National University of Singapore as an assistant professor. His main research interests include nonparametric and semiparametric statistics, random forests, causal inference, precision medicine, survival analysis, generalized fiducial inference, and foundations of statistics.



Min-ge Xie receives his Ph.D. from University of Illinois at Urbana-Champaign in 1996. He is Distinguished Professor and Director of the Office of Statistical Consulting in Department of Statistics, Rutgers, The State University of New Jersey. His main research interest is in the foundation of statistical inference and fusion learning. His other expertise includes estimating equations, hierarchical models, asymptotics and applications in biostatistics and industry.



Wei-Yin Loh 

Contents

30.1	Introduction	593
30.2	Fitting OLR Models	594
30.3	Logistic Regression Trees	596
30.4	Missing Values and Cyclic Variables	598
30.5	Conclusion	602
	References	603

Abstract

Ordinary logistic regression (OLR) models the probability of a binary outcome. A logistic regression tree (LRT) is a machine learning method that partitions the data and fits an OLR model in each partition. This chapter motivates LRT by highlighting the challenges of OLR with respect to model selection, interpretation, and visualization on a completely observed dataset. Being nonparametric, a LRT model typically has higher prediction accuracy than OLR for large datasets. Further, by sharing model complexity between the tree structure and the OLR node models, the latter can be made simple for easier interpretation and visualization.

OLR is more challenging if there are missing values in the predictor variables, because imputation must be carried out first. The second part of the chapter reviews the GUIDE method of constructing LRT models. A strength of GUIDE is its ability to deal with large numbers of variables and without the need to impute missing values. This is demonstrated on a vehicle crash-test dataset for which imputation is difficult due to missing values and other problems.

Keywords

Classification and regression trees · Imputation · Logistic regression · Machine learning · Missing data · Visualization

30.1 Introduction

Ordinary logistic regression (OLR) is a technique for modeling the probability of a binary outcome in terms of one or more predictor variables. Consider, for example, a dataset on tree damage during a severe thunderstorm over 477,000 acres of the Boundary Waters Canoe Area Wilderness in northeastern Minnesota in July 4, 1999 (R package `alr4` [1]). Observations from 3666 trees were collected, including for each tree, whether it was blown down ($Y = 1$) or not ($Y = 0$), its trunk diameter D in centimeters, its species S , and the local intensity L of the storm, as measured by the fraction of damaged trees in its vicinity.

Let $p = P(Y = 1)$ denote the probability that a tree is blown down. OLR approximates the logit function $\text{logit}(p) = \log(p/(1-p))$ as a function of the predictor variables linear in any unknown parameters. A *simple linear* OLR model has the form $\text{logit}(p) = \log(p/(1-p)) = \beta_0 + \beta_1 X$, where X is the only predictor variable. Solving for p yields the p -function

$$p = \frac{\exp(\beta_0 + \beta_1 X)}{1 + \exp(\beta_0 + \beta_1 X)} = \frac{1}{1 + \exp(-\beta_0 - \beta_1 X)}$$

In general, if there are k predictor variables, X_1, \dots, X_k , a *multiple linear* OLR model has the form $\text{logit}(p) = \beta_0 + \sum_{j=1}^k \beta_j X_j$. The parameters $\beta_0, \beta_1, \dots, \beta_k$ are typically estimated by maximizing the likelihood function. Let n denote the sample size, and let $(x_{i1}, \dots, x_{ik}, y_i)$ denote the values of (X_1, \dots, X_k, Y) for the i th observation ($i = 1, \dots, n$). Treating each y_i as the outcome of an independent Bernoulli random variable with success probability p_i , the likelihood function is

W.-Y. Loh (✉)

Department of Statistics, University of Wisconsin, Madison, WI, USA
e-mail: loh@stat.wisc.edu

$$\prod_{i=1}^n p_i^{y_i} (1 - p_i)^{1-y_i} = \frac{\exp\{\sum_i y_i(\beta_0 + \sum_j \beta_j x_{ij})\}}{\prod_i \{1 + \exp(\beta_0 + \sum_j \beta_j x_{ij})\}}$$

The maximum likelihood estimates are the values of $(\beta_0, \beta_1, \dots, \beta_k)$ that maximize this function.

30.2 Fitting OLR Models

Fitting a simple linear OLR model to the tree damage data using L yields

$$\text{logit}(p) = -1.999 + 4.407L \tag{30.1}$$

with estimated p -function shown in Fig. 30.1. The equation implies that the stronger the local storm intensity, the higher the chance that a tree is blown down. The boxplots in Fig. 30.2 show that the distributions of D are skewed. To reduce the skewness, Cook and Weisberg [2] transformed D to $\log(D)$ and obtained the model

$$\text{logit}(p) = -4.792 + 1.749 \log(D) \tag{30.2}$$

which suggests that larger trees are less likely to survive the thunderstorm than narrower ones. If both $\log(D)$ and L are used, the model becomes

$$\text{logit}(p) = -6.677 + 1.763 \log(D) + 4.42L. \tag{30.3}$$

The relative stability of the coefficients of L and $\log(D)$ in Eqs. (30.1)–(30.3) is due to the weak correlation of 0.168 between the two variables. If the interaction $L \log(D)$ is included, the model changes to

$$\text{logit}(p) = -4.341 + 0.891 \log(D) - 1.482L + 2.235L \log(D) \tag{30.4}$$

and the coefficients of $\log(D)$ and L are changed more dramatically.

So far, species S has been excluded from the models. As in linear regression, a categorical variable having m distinct values may be represented by $(m - 1)$ indicator variables, U_1, \dots, U_{m-1} , each taking value 0 or 1. The variables for species are shown in Table 30.1, which uses the “set-to-zero constraint” that sets all the indicator variables to 0 for the first species (aspen). A model that assumes the same slope coefficients for all species but that gives each a different intercept term is

$$\begin{aligned} \text{logit}(p) = & -5.997 + 1.581 \log(D) + 4.629L \\ & - 2.243U_1 + 0.0002U_2 + 0.167U_3 - 2.077U_4 \\ & + 1.040U_5 - 1.724U_6 - 1.796U_7 \\ & - 0.003U_8. \end{aligned} \tag{30.5}$$

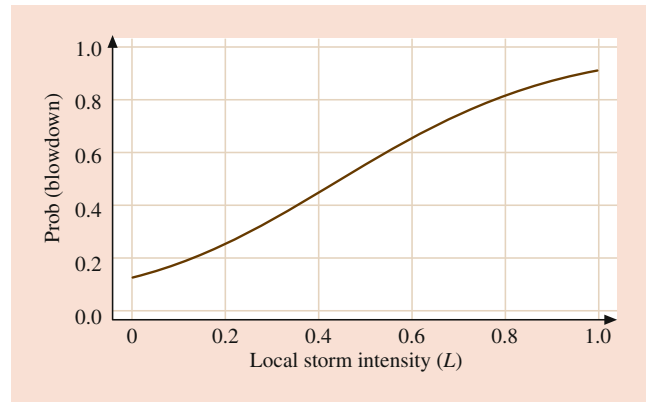


Fig. 30.1 Estimated probability of blowdown computed from a simple linear logistic regression model using L as predictor

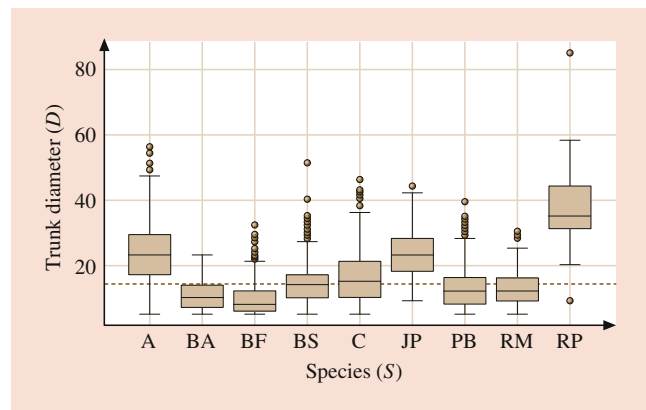


Fig. 30.2 Boxplots of trunk diameter D . The dotted line marks the median value of D

Table 30.1 Indicator variable coding for species variable S

Species	U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8
A (aspen)	0	0	0	0	0	0	0	0
BA (black ash)	1	0	0	0	0	0	0	0
BF (balsam fir)	0	1	0	0	0	0	0	0
BS (black spruce)	0	0	1	0	0	0	0	0
C (cedar)	0	0	0	1	0	0	0	0
JP (jack pine)	0	0	0	0	1	0	0	0
PB (paper birch)	0	0	0	0	0	1	0	0
RM (red maple)	0	0	0	0	0	0	1	0
RP (red pine)	0	0	0	0	0	0	0	1

How well do models (30.1)–(30.5) fit the data? One popular way to assess fit is by means of significance tests based on the residual deviance and its degrees of freedom (df)—see, e.g., [3, p. 96] for the definitions. The residual deviance is analogous to the residual sum of squares in linear regression. For model (30.5), the residual deviance is 3259 with 3655 df. We can evaluate the fit of this model by comparing its residual deviance against that of a larger one, such as the 27-parameter model

$$\begin{aligned} \text{logit}(p) &= \beta_0 + \beta_1 \log(D) + \beta_2 L + \sum_{j=1}^8 \gamma_j U_j \\ &+ \sum_{j=1}^8 \beta_{1j} U_j \log(D) + \sum_{j=1}^8 \beta_{2j} U_j L \end{aligned} \quad (30.6)$$

that allows the coefficients of $\log(D)$ and L to vary with species. It has a residual deviance of 3163 with 3639 df. If model (30.5) fits the data well, the difference between its residual deviance and that of model (30.6) is approximately distributed as a chi-squared random variable with df equal to the difference in df of the two models. The difference in deviance is $3259 - 3163 = 96$, which is improbably large for a chi-squared random variable with $3655 - 3639 = 16$ df.

Rejection of model (30.5) does not necessarily imply that model (30.6) is satisfactory. To find out, it may be compared with a larger one, such as the 28-parameter model

$$\begin{aligned} \text{logit}(p) &= \beta_0 + \beta_1 \log(D) + \beta_2 L + \beta_3 L \log(D) + \sum_{j=1}^8 \gamma_j U_j \\ &+ \sum_{j=1}^8 \beta_{1j} U_j \log(D) + \sum_{j=1}^8 \beta_{2j} U_j L \end{aligned} \quad (30.7)$$

that includes an interaction between L and $\log(D)$. This has a residual deviance of 3121 with 3638 df. Therefore model (30.6) is rejected because its residual deviance differs from that of (30.7) by 42 but their dfs differ only by 1. With this procedure, each of models (30.1) through (30.6) is rejected when compared against the next larger model in the sequence.

Another way to select a model employs a function such as AIC, which is residual deviance plus two times the number of estimated parameters. AIC tries to balance deviance against model complexity (see, e.g., [4, p. 234]), but it tends to overfit the data. That is, AIC often chooses a large model. In this dataset, if we apply AIC to the set of all models up to third order, it chooses the largest, namely, the three-factor interaction model

$$\begin{aligned} \text{logit}(p) &= \beta_0 + \beta_1 \log(D) + \beta_2 L + \sum_{j=1}^8 \gamma_j U_j \\ &+ \beta_3 L \log(D) + \sum_{j=1}^8 \beta_{1j} U_j \log(D) \\ &+ \sum_{j=1}^8 \beta_{2j} U_j L + \sum_{j=1}^8 \delta_j U_j L \log(D) \end{aligned} \quad (30.8)$$

which has 36 parameters.

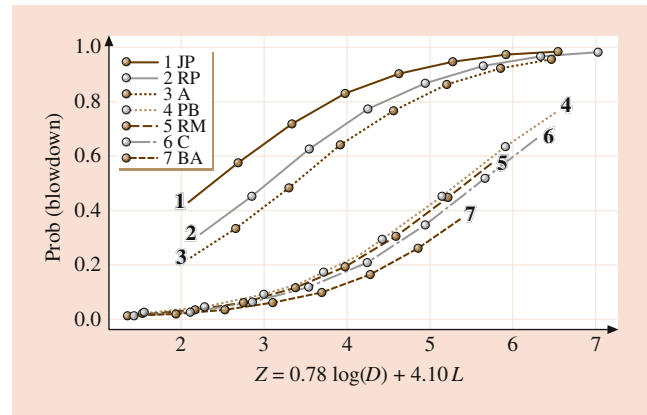


Fig. 30.3 Estimated probability of blowdown for seven species, excluding balsam fir (BF) and black spruce (BS), according to model (30.9)

Models (30.7) and (30.8) are hard to graph. Plotting the estimated p -function as in Fig. 30.1 is impossible if a model has more than one predictor variable. This problem is exacerbated by the tendency of model complexity increasing with increase in sample size and number of predictors. Interpretation of the estimated coefficients is futile then, as they often change from one model to another, due to multicollinearity among the terms. For example, the coefficient for L is 4.424, -1.482 , and 4.629 in models (30.3), (30.4), and (30.5), respectively.

To deal with this problem, [2] used a “partial one-dimensional model” (POD) that employs a linear function of $\log(D)$ and L as predictor variable. They found that if the observations for balsam fir (BF) and black spruce (BS) are excluded, the model $\text{logit}(p) = \beta_0 + Z + \sum_j \gamma_j U_j$, with $Z = 0.78 \log(D) + 4.1L$, fits the remaining data quite well. Now the estimated p -function can be plotted as shown in Fig. 30.3, but the graph is not as simple to interpret as that in Fig. 30.1 because Z is a linear combination of two variables. To include species BF and BS, [2] settled on the larger model

$$\begin{aligned} \text{logit}(p) &= \beta_0 + Z + \sum_{j=1}^9 \gamma_j U_j + (\theta_1 I_{\text{BF}} + \theta_2 I_{\text{BS}}) \log(D) \\ &+ (\phi_1 I_{\text{BF}} + \phi_2 I_{\text{BS}}) L \end{aligned} \quad (30.9)$$

which contains separate coefficients (θ_j, ϕ_j) for BF and BS. Here $I_{(\cdot)}$ denotes the indicator function, i.e., $I_A = 1$ if species is A, and $I_A = 0$ otherwise. The model cannot be displayed graphically for species BF and BS because it is a function of three predictor variables.

30.3 Logistic Regression Trees

A logistic regression tree (LRT) model is a machine learning solution that simultaneously retains the graphical advantage of simple models and the prediction accuracy of more complex ones. It recursively partitions the dataset and fits a simple or multiple linear OLR model in each partition. As a result, the partitions can be displayed as a decision tree [5] such as Fig. 30.4, which shows a *simple linear* LRT model fitted to the tree damage data by the GUIDE algorithm [6, 7]. A terminal node represents a partition, and an OLR model with a single linear predictor is fitted in each one. Beside each intermediate node is a condition stating that an observation goes to the left subnode if and only if the condition is satisfied. Below each terminal node are the sample size (in italics), the proportion of blown down trees, and the name of the best linear predictor variable. The split at the root node (labeled “1”) sends observations to node 2 if and only if S is A, BS, JP, or RP. (Node labels employ the convention that a node with label k has left and right child nodes labeled $2k$ and $2k + 1$, respectively.) Node 5, consisting of the JP and RP species, has the highest proportion of blown down trees at 0.82. Node 9, which consists of species A and BS trees with diameters greater than 9.75 cm, has the second highest proportion of 0.67. Variable L is the best linear predictor in all terminal nodes except nodes 13 and 15, where D is the best linear predictor. The main advantage in using one linear predictor in each node is that the fitted p -functions can be displayed graphically, as shown in Fig. 30.5. It is not necessary to transform D to $\log(D)$ in the LRT.

The LRT model in Fig. 30.4 may be considered a different kind of POD model from that proposed in [2]. Whereas the

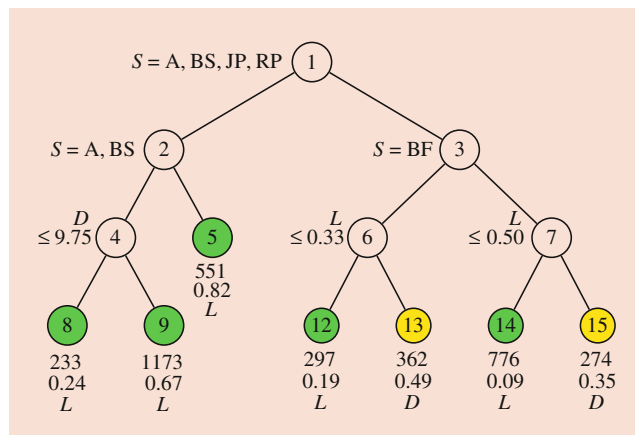


Fig. 30.4 GUIDE simple linear LRT model for $P(\text{blowdown})$. At each split, an observation goes to the left branch if and only if the condition is satisfied. Sample size (in italics), proportion of blowdowns, and name of regressor variable are printed beneath each terminal node. Green and yellow terminal nodes have L and D , respectively, as best linear predictor

word “partial” in POD refers to model (30.9) being one-dimensional if restricted to certain parts of the data (species in this example), it refers to partitions of the predictor space in a LRT. In addition, whereas “one-dimensional” refers to Z being a linear combination of $\log(D)$ and L in (30.9), the OLR predictor in each node of a LRT is trivially one-dimensional because it is an original variable.

GUIDE is a classification and regression tree algorithm with origins in the FACT [8], SUPPORT [9], QUEST [10], CRUISE [11, 12], and LOTUS [13] methods; see [14]. All of them split a dataset recursively, choosing a single X variable to split each node. If X is an ordinal variable, the split typically has the form $s = \{X \leq c\}$, where c is a constant. If X is a categorical variable, the split has the form $s = \{X \in \omega\}$, where ω is a subset of the values taken by X . For linear regression trees, algorithms such as AID [15], CART [16], and M5 [17] choose s to minimize the total sum of squared residuals of the regression models fitted to the two data subsets formed by s . Though seemingly innocuous, this approach is flawed as it is biased toward choosing X variables that allow more splits. To see this, suppose that X is an ordinal variable having m distinct values. Then there are $(m-1)$ ways to split the data along the X axis, with each split $s = \{X \leq c\}$ being such that c is the midpoint between two consecutively ordered distinct values of X . This creates a *selection bias* toward X variables with large values of m . In the current example, variable L has 709 unique values but D has only 87. Hence L has eight times as many opportunities as D to split the data. The bias is worse if there are high-level categorical variables, because a categorical variable having m categorical values permits $(2^{m-1} - 1)$ splits of the form $s = \{X \in \omega\}$. For example, variable S permits $(2^{9-1} - 1) = 255$ splits, which is almost three times as many splits as D allows. The earliest warning on the potential for the bias to produce misleading conclusions seems to be [18].

GUIDE avoids the bias by using a two-step approach to split selection. First, it uses significance tests to select the X variable. Then it searches for c or ω for X . For linear regression trees, this is achieved by fitting a linear model to the data in the node and using a contingency table chi-squared test of the association between grouped values of each predictor variable and the signs of the residuals. If X is ordinal, the groups are intervals between certain order statistics. If X is categorical, the groups are the categorical levels. Then the X variable having the smallest chi-squared p -value is selected. Repeating this procedure recursively produces a large binary tree that is pruned to minimize a cross-validation estimate of prediction mean squared error [16].

Let $\hat{p}(x)$ denote the estimated value of $p(x) = P(Y = 1 | X = x)$. The preceding split variable selection method needs modification for logistic regression, because the residual $y - \hat{p}(x)$ is positive if $y = 1$ and negative if $y = 0$, irrespective of the value of $\hat{p}(x)$. Consequently, the residual

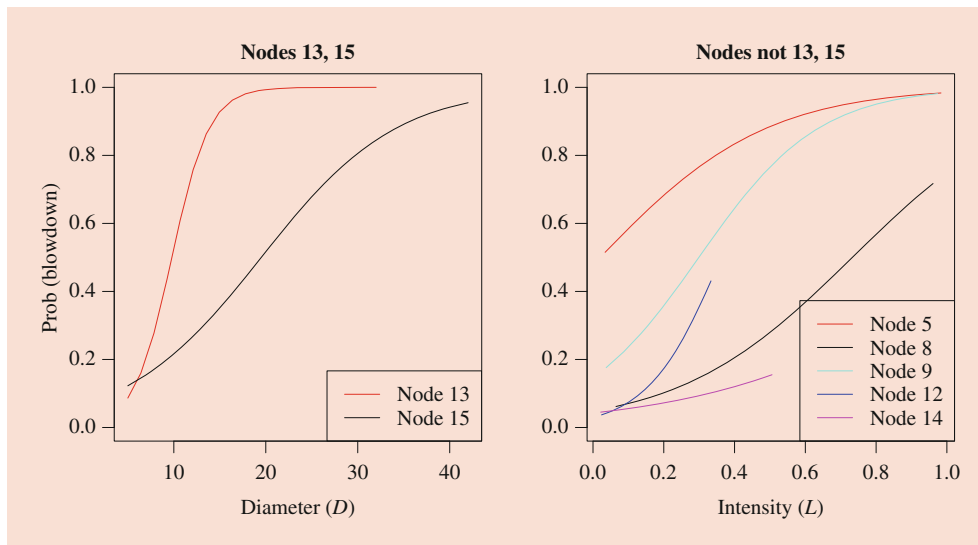


Fig. 30.5 Estimated p -functions in terminal nodes of the tree in Fig. 30.4

signs provide no information on the adequacy of $\hat{p}(x)$. A first attempt at a solution was proposed in [19], where the residuals $y - \hat{p}(x)$ are replaced with “pseudo-residuals” $\bar{p}(x) - \hat{p}(x)$, with $\bar{p}(x)$ being a weighted average of the y values in a neighborhood of x . Its weaknesses are sensitivity to choice of weights and neighborhoods and difficulty in specifying the neighborhoods if the dimension of the predictor space is large or if there are missing values. LOTUS uses a trend-adjusted chi-squared test [20, 21] that effectively replaces $\bar{p}(x)$ with a linear estimate.

For logistic regression, GUIDE uses the average from an ensemble of least-squares GUIDE regression trees (called a “GUIDE forest”) to form the pseudo-residuals for variable selection. The main steps are as follows:

1. Fit a least-squares GUIDE forest [22] to the data to obtain a preliminary estimate $\bar{p}(x)$ of $p(x)$ for each observed x . (Random forest [23] cannot substitute for GUIDE forest if the data contain missing values.)
2. Beginning with the root node, carry out the following steps on the data in each node, stopping only if the number of observations is below a pre-specified threshold or if all the values of the predictor variables or the Y values are constant:
 - (a) For each X variable to be used in fitting an OLR model in the node, temporarily impute its missing values with its node sample mean.
 - (b) Fit a simple or multiple linear OLR model to the imputed data in the node. If a simple linear OLR model is desired, fit one to each linear predictor variable in turn, and choose the one with smallest residual deviance.

Let $\hat{p}(x)$ denote the estimated value of $p(x)$ from the fitted model.

- (c) Revert the imputed values in step (2a) to their original missing state.
- (d) For each ordinal X variable, let $q_1 \leq q_2 \leq q_3$ denote its sample quartiles at the node, and define the categorical variable $V = \sum_{j=1}^3 I(X > q_j)$. If X is a categorical variable, define $V = X$. Add an extra “missing” category to V if X has missing values.
- (e) Form a contingency table for each X variable using the signs of $\bar{p}(x) - \hat{p}(x)$ as rows and the values of V as columns. Find the chi-squared statistic χ_v^2 for the test of independence between rows and columns.
- (f) Let $G_v(x)$ denote the distribution function of a chi-squared variable with v df, and let $\epsilon = 2 \times 10^{-6}$. Convert each χ_v^2 to its equivalent one-df χ_1^2 value as follows:
 - i. If $\epsilon < G_v(\chi_v^2) < 1 - \epsilon$, define $\chi_1^2 = G_1^{-1}(G_v(\chi_v^2))$.
 - ii. Otherwise, to avoid dealing with very small or large p-values, use the following dual application of the Wilson-Hilferty approximation [24]. Define

$$W_1 = \left\{ \sqrt{2\chi_v^2} - \sqrt{2v-1} + 1 \right\}^2 / 2$$

$$W_2 = \max \left(0, \left[\frac{7}{9} + \sqrt{v} \left\{ \left(\frac{\chi_v^2}{v} \right)^{1/3} - 1 + \frac{2}{9v} \right\}^3 \right] \right).$$

Approximate the one-df chi-squared value with

$$\chi_1^2 = \begin{cases} W_2 & \text{if } \chi_v^2 < v + 10\sqrt{2v} \\ (W_1 + W_2)/2 & \text{if } \chi_v^2 \geq v + 10\sqrt{2v} \text{ and } \\ & W_2 < \chi_v^2 \\ W_1 & \text{otherwise.} \end{cases}$$

An earlier one-step approximation is used in [7]. Tables 30.2 and 30.3 show the contingency tables and corresponding chi-squared statistics for Species, Intensity, and Diameter at the root node of the tree in Fig. 30.4.

- (g) Let X^* be the variable with the largest value of χ_1^2 , and let NA denote the missing value code:
 - i. If X^* is ordinal, let s be a split of the form $\{X^* = \text{NA}\}$, $\{X^* \leq c\} \cup \{X^* = \text{NA}\}$, or $\{X^* \leq c\} \cap \{X^* \neq \text{NA}\}$.
 - ii. If X^* is categorical, let s be a split of the form $\{X^* \in \omega\}$, where ω is a proper subset of the values (including NA) of X^* .
 - (h) For each split s , apply steps (2a) and (2b) to the data in the left and right subnodes induced by s , and let $d_L(s)$ and $d_R(s)$ be their respective residual deviances.
 - (i) Select the split s that minimizes $d_L(s) + d_R(s)$.
3. After splitting stops, prune the tree with the CART cost-complexity method [16] to obtain a nested sequence of subtrees.
 4. Use the CART cross-validation method to estimate the prediction deviance of each subtree.
 5. Select the smallest subtree whose estimated prediction deviance is within a half standard error of the minimum.

Figure 30.6 shows the LOTUS tree for the current data. MOB [25] is another algorithm that can construct a LRT, but for simple linear LRT models, it requires the linear predictor to be pre-specified and to be the same in all terminal nodes. Figure 30.7 shows the MOB tree with L as the common

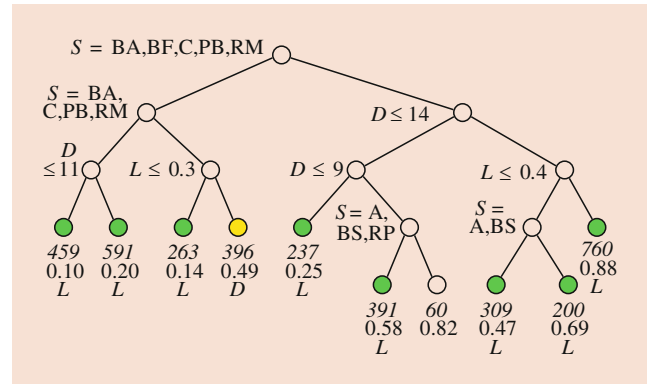


Fig. 30.6 LOTUS simple linear LRT model for P(blowdown). At each split, an observation goes to the left branch if and only if the condition is satisfied. Sample size (in italics), proportion of blowdowns, and name of regressor variable (if any) are printed below nodes. Green and yellow terminal nodes have L and D, respectively, as best linear predictor

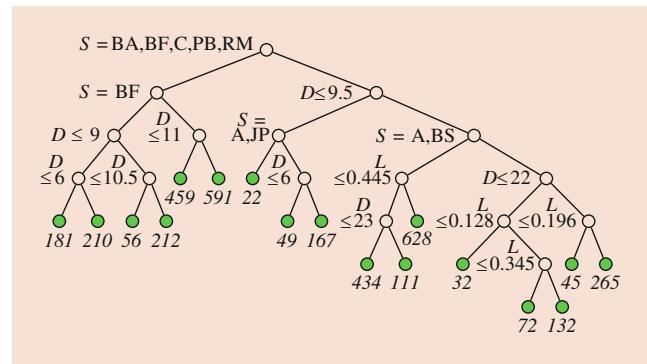


Fig. 30.7 MOB simple linear LRT model with L pre-specified as the common linear predictor in all nodes. At each split, an observation goes to the left branch if and only if the condition is satisfied. Sample sizes (in italics) are printed below nodes

Table 30.2 Chi-squared test for Species with Wilson-Hilferty χ_1^2 value

	A	BA	BF	BS	C	JP	PB	RM	RP
$\tilde{p} > \hat{p}$	413	0	239	673	0	501	0	2	47
$\tilde{p} \leq \hat{p}$	23	75	420	297	355	1	497	121	2
	$\chi_8^2 = 2125, \chi_1^2 = 1942$								

Table 30.3 Chi-squared tests for Intensity and Diameter with quartile intervals Q_1, Q_2, Q_3, Q_4 and Wilson-Hilferty χ_1^2 values

	Intensity				Diameter			
	Q_1	Q_2	Q_3	Q_4	Q_1	Q_2	Q_3	Q_4
$\tilde{p} > \hat{p}$	327	418	527	603	14	543	637	681
$\tilde{p} \leq \hat{p}$	595	493	390	313	933	424	281	153
	$\chi_3^2 = 195, \chi_1^2 = 171$				$\chi_3^2 = 1378, \chi_1^2 = 1314$			

linear predictor. Figure 30.8 compares the values of $\hat{p}(x)$ from a GUIDE forest of 500 trees, model (30.9) and the simple linear GUIDE, LOTUS, and MOB LRT models. Although there are clear differences in the values of $\hat{p}(x)$ between GUIDE, LOTUS, and MOB, they seem to compare similarly against (30.9) and GUIDE forest. Figure 30.9 shows the corresponding results where LOTUS fits the multiple linear LRT model $\text{logit}(p) = \beta_0 + \beta_1 D + \beta_2 L$ and GUIDE and MOB fit $\text{logit}(p) = \beta_0 + \beta_1 D + \beta_2 L + \sum_{j=1}^8 \gamma_j U_j$ in each terminal node. (LOTUS does not convert categorical variables to indicator variables to serve as regressors.) The correlations among the $\hat{p}(x)$ values are much higher.

30.4 Missing Values and Cyclic Variables

The US National Highway Traffic Safety Administration has been evaluating vehicle safety by performing crash tests with dummy occupants since 1972 ([ftp://www.nhtsa.dot.gov/ges](http://www.nhtsa.dot.gov/ges)).

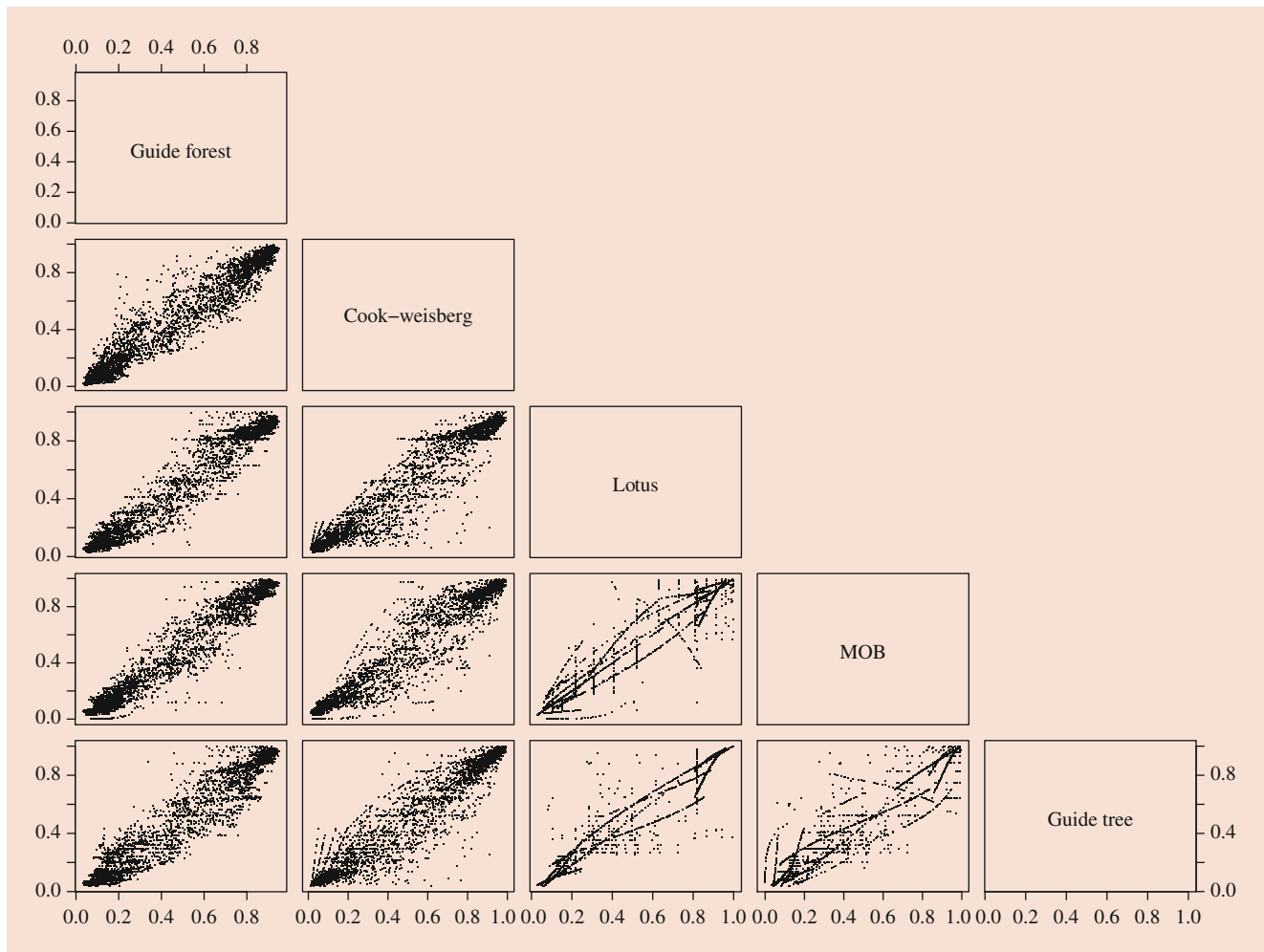


Fig. 30.8 Comparison of fitted values \hat{p} of Cook-Weisberg model (30.9) and GUIDE forest versus simple linear LRT models

We use data from 3310 crash tests where the test dummy is in the driver's seat to show how GUIDE deals with missing values and cyclic variables. Each test gives the severity of head injury (HIC) sustained by the dummy and the values of about 100 variables describing the vehicle, test environment, and the test dummy. The response variable is $Y = 1$ if $HIC > 1000$ (threshold for severe head injury) and $Y = 0$ otherwise. About half of the predictor variables are ordinal, six are cyclic, and the rest are categorical.

Three features in the data make model building particularly challenging. The first is missing data. Missing values in categorical variables are not problematic, as they can be assigned a "missing" category. Missing values in other variables, however, need to be imputed before application of OLR. This can be extraordinarily difficult if there are many missing values and the missingness patterns are complex [22, 26]. All ordinal and cyclic variables here have missing values. Table 30.4 gives the names and numbers of missing values of some of them (see [27] for the others). For example, IMPANG, the angle between the axis of a vehicle and the

axis of another vehicle or barrier, is undefined for a rollover crash test, where there is no barrier and only one vehicle is involved. In such cases, the value of IMPANG is recorded as missing and imputing it with a number is inappropriate. The situation is worse for variable CARANG, which has 991 missing values. Given that the crash tests are carefully monitored and have been performed for years, it is unlikely for so many observations to be missing by chance.

For split selection, GUIDE sends all missing values in the selected ordinal or cyclic variable either to the left or to the right subnode, depending on which split gives a smaller sum of residual deviances in the two subnodes. Hence no imputation is carried out in this step. To fit an OLR model to a node, GUIDE imputes missing values in the selected predictor variable with its node mean.

A second challenging feature is the presence of cyclic variables that are angles with periods of 360 degrees. These variables are traditionally transformed to sines and cosines, but splits on one of them at a time are not as meaningful as splits on the angles themselves. The problem is more

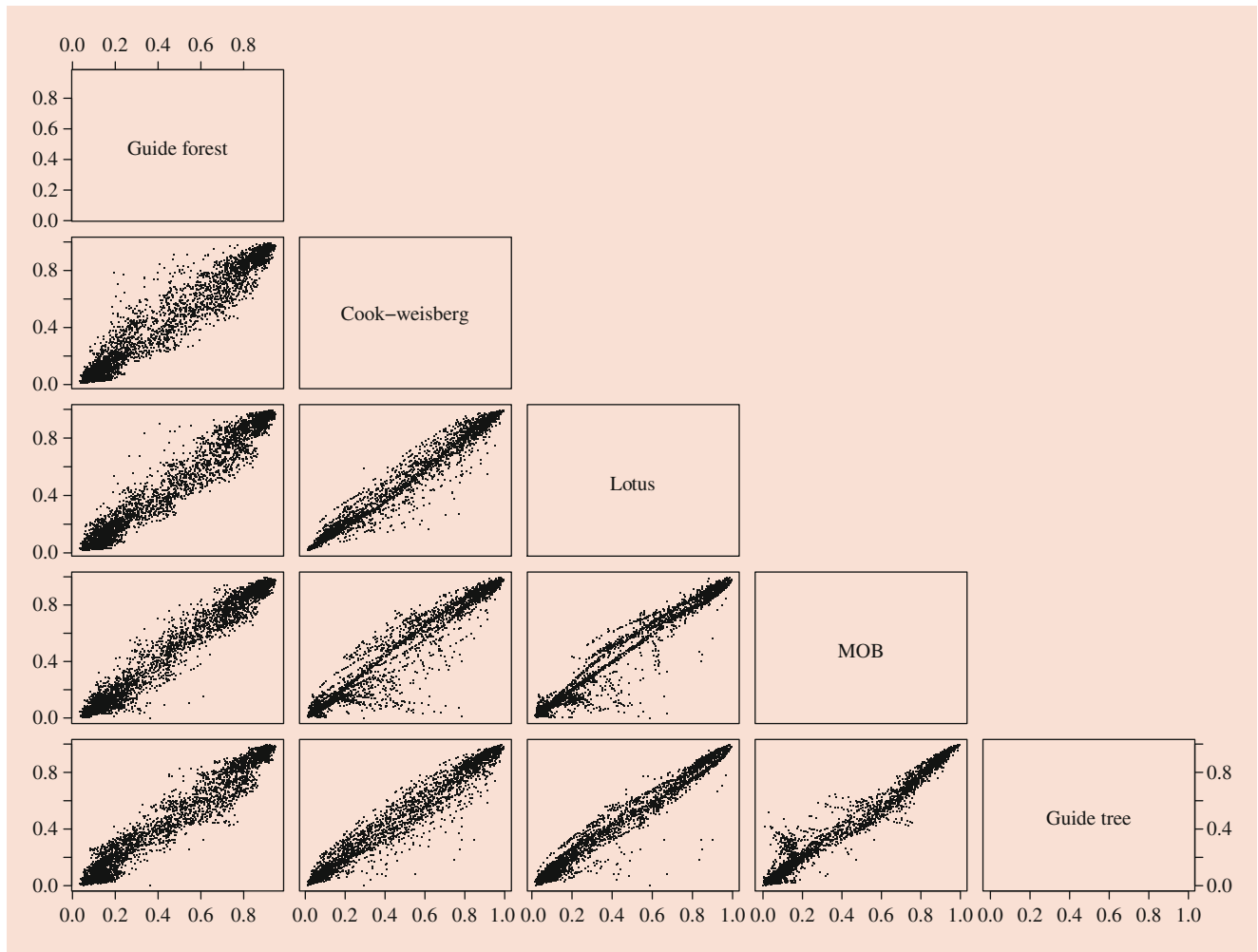


Fig. 30.9 Comparison of fitted values \hat{p} of Cook-Weisberg model (30.9) and GUIDE forest versus multiple linear LRT models

Table 30.4 Definitions and numbers of missing values of some predictor variables in the crash-test data

Variable	Description	Missing
BARRIG	Rigid or deformable barrier	1
BARSHP	Barrier shape (21 types)	0
BX2	Rear surface of vehicle to front of engine	288
BX5	Rear surface of vehicle to upper leading edge of left door	288
CARANG	Angle between surface of rollover test cart and ground	991
COLMEC	Steering column collapse mechanism (9 types)	248
ENGDSP	Engine displacement	24
IMPANG	Angle between axis of vehicle 2 and axis of vehicle 1 or barrier (0 degree is perpendicular to barrier)	4
CLSSPD	Closing speed: relative velocity of approach of two centers of gravity before contact	2
VEHSPD	Resultant speed of vehicle before impact	1
VEHTWT	Vehicle test weight	4
VEHWID	Vehicle width	90
WHLBAS	Vehicle or impactor's wheelbase	30
YEAR	Vehicle model year	4

difficult if the variable has missing values. Should we impute the angles and then compute the sines and cosines of the imputed values, or should we impute the sines and cosines

directly? GUIDE avoids imputation entirely by restricting cyclic variables to split the nodes. If a cyclic variable is selected, the split takes the form of a sector “ $X \in [\theta_1, \theta_2]$,”

where θ_1 and θ_2 are angles, and missing values are sent to the left or right subnode in the same fashion as noncyclic variables.

The third challenging feature is that, apparently by design, high-speed crash tests are more often carried out on deformable barriers and low-speed tests more often on rigid

barriers. This is evident from the boxplots of CLSSPD by BARRIG in Fig. 30.10, where half of the tests with deformable barriers are above closing speeds of 60 km/h, but less than one quarter of those with rigid barriers are above 60 km/h. Presumably, crashes into rigid barriers are not performed at high speeds because the outcomes are predictable, but this confounds the effects of CLSSPD and BARRIG in an OLR model.

We say that X is an “s” variable if it can be used to *split* the nodes and an “f” variable if it can be used to *fit* OLR models in the nodes. To limit the amount of imputation in this example, we restrict ordinal variables with more than 20 percent missing values to serve as s variables only. Cyclic and categorical variables are also restricted to splitting nodes.

Figure 30.11 shows the LRT where a simple linear OLR model is fitted in each node. The root node is split on COLMEC, which is steering wheel collapse mechanism. Observations with COLMEC equal to BWU (behind wheel unit), EMB (embedded ball), EXA (extruded absorber), NON (none), or OTH (other) go to node 2. Otherwise, if COLMEC is CON (convoluted tube), CYL (cylindrical mesh tube), NAP (not applicable), UNK (unknown), or missing, observations go to node 3. At node 2, observations go to node 4 if $BX2 \leq 3496.5$ or missing (the asterisk beside the inequality sign in the figure indicates that missing values go to the left node). At node 3,

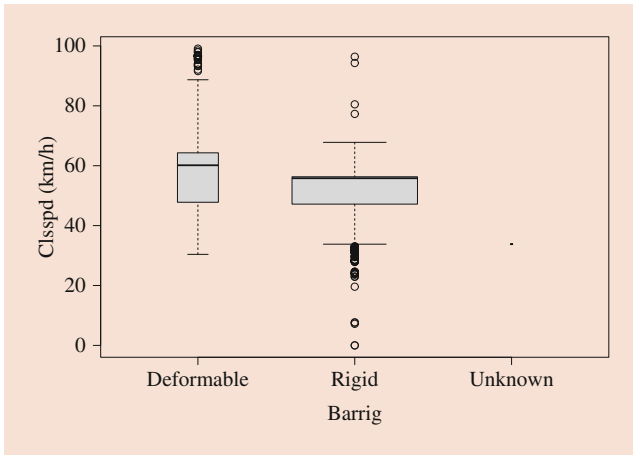


Fig. 30.10 Boxplots of closing speed by barrier rigidity for the crash-test data, with box width proportional to square root of sample size

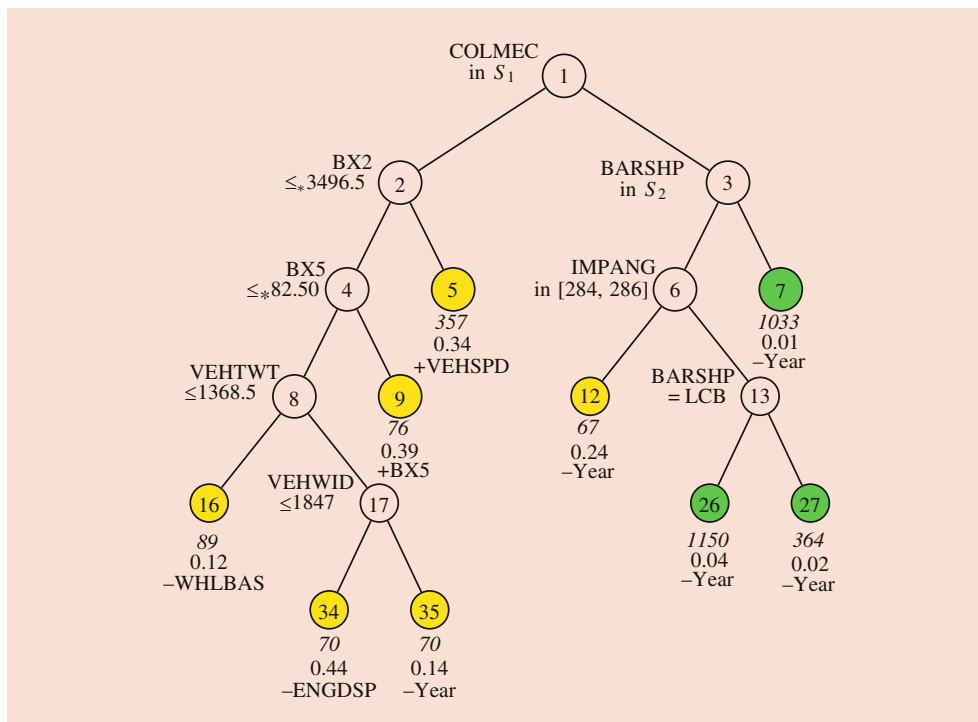


Fig. 30.11 GUIDE piecewise simple linear LRT for crash-test data. At each split, an observation goes to the left branch if and only if the condition is satisfied. The symbol “ \leq_* ” stands for “ \leq or missing.” Set $S_1 = \{BWU, EMB, EXA, NON, OTH\}$. Set $S_2 = \{LCB, POL, US2, US3\}$.

Sample size (*in italics*), proportion of cases with $Y = 1$, and sign and name of regressor variable printed below nodes. Terminal nodes with proportions of $Y = 1$ above and below value of 0.08 at root node are colored yellow and green, respectively

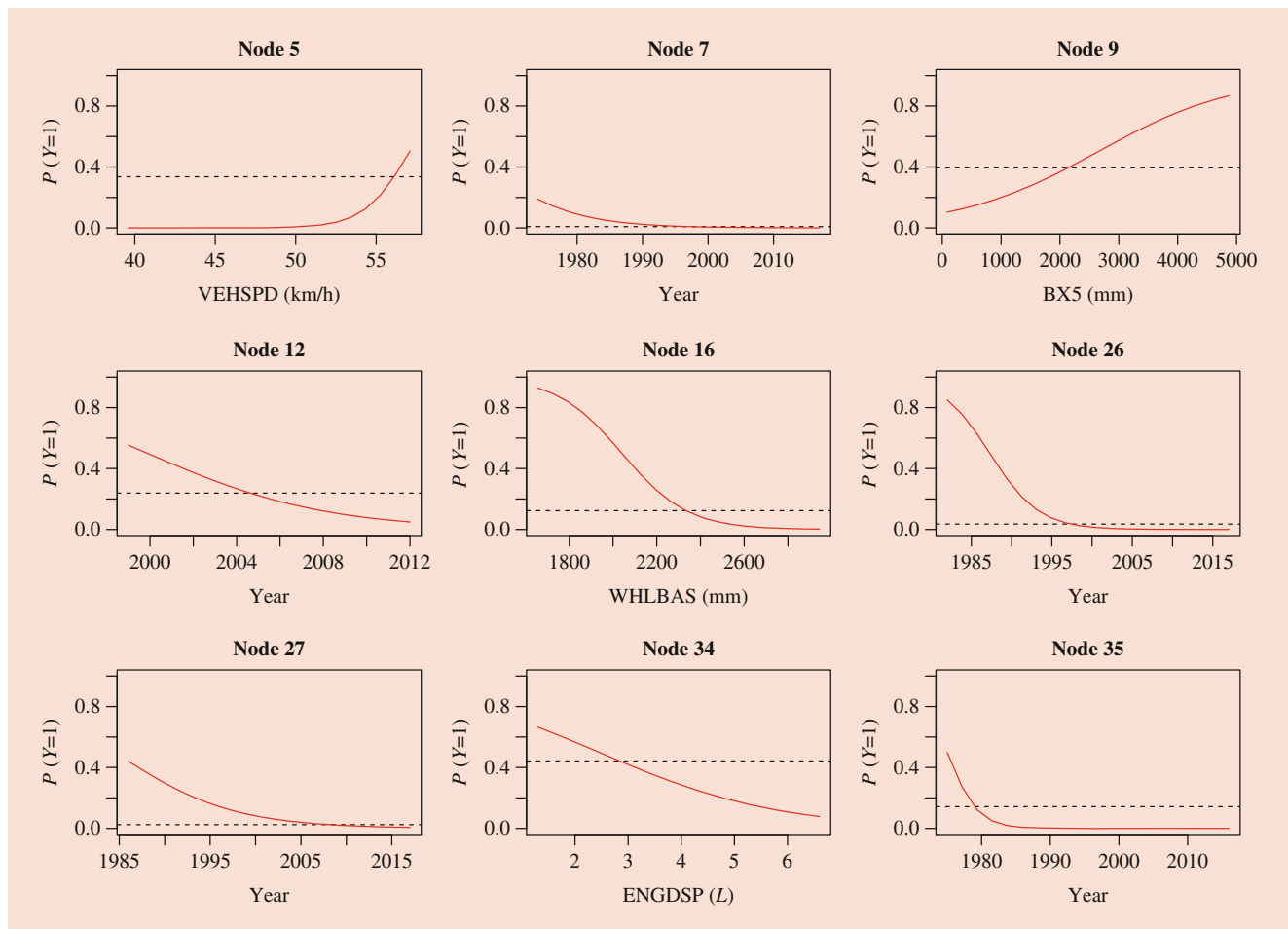


Fig. 30.12 Fitted logistic regression curves in terminal nodes of Fig. 30.11; horizontal dotted lines indicate proportion of severe injury in the node

observations go to node 6 if BARSHP is LCB (load cell barrier), POL (pole), US2, or US3 (different barrier types). Node 6 is split on impact angle IMPANG, where 0 degree indicates impact is head-on. If an observation has IMPANG between 284 and 286 degrees inclusive (i.e., Driver side), it goes to node 12. The 2-degree range may seem narrow, but there are 67 observations in the node, suggesting that the tests were by design. Below each terminal node are the sample size (in *italics*), proportion of $Y = 1$, and the selected OLR predictor variable, with the sign of its estimated coefficient.

The tree shows that nodes 5, 9, and 34 have the highest proportions of severe head injury, at 34, 39, and 44%, respectively. Vehicles in these nodes have certain steering wheel collapse mechanisms, and they tend to be longer ($BX2 > 3496.5$ or $BX5 > 82.5$) or are heavy ($VEHTWT > 1368.5$) and narrow ($VEHWID \leq 1846$). Figure 30.12 shows the fitted logistic regression curves in the terminal nodes. The proportion of tests with severe head injury is indicated by a dotted line in each plot.

30.5 Conclusion

Logistic regression is a technique for estimating the probability of an event in terms of the values of one or more predictor variables. If there are missing values among the predictor variables, they need to be imputed first. Otherwise, the observations or variables containing the missing values would need to be deleted. Neither solution is attractive. In practice, finding a logistic regression model with good prediction accuracy is seldom automatic; it usually requires trial-and-error selection of variables, choice of transformations, and estimation of the accuracy of numerous models. Even when a model with good estimated accuracy is found, interpretation of the regression coefficients is not straightforward if there are two or more predictor variables.

A logistic regression tree is a piecewise logistic regression model, with the pieces obtained by recursively partitioning the space of predictor variables. Consequently, if there

is no over-fitting, it may be expected to possess higher prediction accuracy than a one-piece logistic regression model. Recursive partitioning has two advantages over a search of all partitions: it is computationally efficient and it allows the partitions to be displayed as a decision tree. At a minimum, a logistic regression tree can serve as an informal goodness of fit test of whether a one-piece logistic model is adequate for the whole sample. A nontrivial pruned tree would indicate that a one-piece logistic model has lower prediction accuracy, possibly due to unaccounted interactions or nonlinearities among the variables. Ideally, an effective tree-growing and pruning algorithm would automatically account for the overlooked effects, making it unnecessary to specify interaction and higher-order terms. It would also allow the models in the terminal nodes to be as simple as desired (such as fitting a single linear predictor in each node).

Tree pruning is very important for prediction accuracy. Many methods adopt the AIC-type approach of selecting the tree that minimizes the sum of the residual deviance and a multiple, K , of the number of terminal nodes. There being no value of K that works for all datasets [16], the advantage of this approach is mainly computational speed. Our experience indicates that it is inferior to a pruning approach that uses cross-validation to estimate prediction accuracy.

Despite a binary decision tree being intuitive to interpret, a poor split selection method can yield misleading conclusions. A common cause is selection bias. The greedy approach used by CART and many other algorithms is known to prefer variables that permit more splits of the data. Consequently, it is hard to know if a variable is chosen due to its predictive power or because it has more ways to partition the data. LOTUS and GUIDE avoid the bias by selecting variables with chi-squared tests. At the time of completion of this article, GUIDE is the only tree algorithm that can deal with cyclic variables and with two or more missing value codes [22]. The GUIDE software and manual may be obtained from www.stat.wisc.edu/~loh/guide.html.

Acknowledgments Part of this work was done in the summer of 2019 during the author's visit to the National Chung Cheng and National Tsing Hua Universities, Taiwan, under the auspices of the National Center for Theoretical Sciences.

References

1. Weisberg, S.: Applied Linear Regression, 4th edn. Wiley, Hoboken, NJ (2014)
2. Cook, R.D., Weisberg, S.: Partial one-dimensional regression models, *American Statistician* **58**, 110–116 (2004)
3. Agresti, A.: An Introduction to Categorical Data Analysis. Wiley, New York (1996)
4. Chambers, J.M., Hastie, T.J.: Statistical Models in S. Wadsworth, Pacific Grove (1992)
5. Loh, W.-Y.: Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **1**, 14–23 (2011). <https://doi.org/10.1002/widm.8>
6. Loh, W.-Y.: Regression trees with unbiased variable selection and interaction detection. *Statistica Sinica* **12**, 361–386 (2002)
7. Loh, W.-Y.: Improving the precision of classification trees. *Ann. Appl. Stat.* **3**, 1710–1737 (2009)
8. Loh, W.-Y., Vanichsetakul, N.: Tree-structured classification via generalized discriminant analysis (with discussion). *J. Am. Stat. Assoc.* **83**, 715–728 (1988)
9. Chaudhuri, P., Huang, M.-C., Loh, W.-Y., Yao, R.: Piecewise-polynomial regression trees. *Statistica Sinica* **4**, 143–167 (1994)
10. Loh, W.-Y., Shih, Y.-S.: Split selection methods for classification trees. *Statistica Sinica* **7**, 815–840 (1997)
11. Kim, H., Loh, W.-Y.: Classification trees with unbiased multiway splits. *J. Am. Stat. Assoc.* **96**, 589–604 (2001)
12. Kim, H., Loh, W.-Y.: Classification trees with bivariate linear discriminant node models. *J. Comput. Graph. Stat.* **12**, 512–530 (2003)
13. Chan, K.-Y., Loh, W.-Y.: LOTUS: An algorithm for building accurate and comprehensible logistic regression trees. *J. Comput. Graph. Stat.* **13**, 826–852 (2004)
14. Loh, W.-Y.: Fifty years of classification and regression trees (with discussion). *Int. Stat. Rev.* **34**, 329–370 (2014)
15. Morgan, J.N., Sonquist, J.A.: Problems in the analysis of survey data, and a proposal. *J. Am. Stat. Assoc.* **58**, 415–434 (1963)
16. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: *Classification and Regression Trees*. Wadsworth, Belmont, CA, USA (1984)
17. Quinlan, J.R.: Learning with continuous classes. In: *Proceedings of AI'92 Australian National Conference on Artificial Intelligence*, pp. 343–348. World Scientific, Singapore (1992)
18. Doyle, P.: The use of Automatic Interaction Detector and similar search procedures. *Oper. Res. Q.* **24**, 465–467 (1973)
19. Chaudhuri, P., Lo, W.-D., Loh, W.-Y., Yang, C.-C.: Generalized regression trees. *Statistica Sinica* **5**, 641–666 (1995)
20. Cochran, W.G.: Some methods of strengthening the common χ^2 tests. *Biometrics* **10**, 417–451 (1954)
21. Armitage, P.: Tests for linear trends in proportions and frequencies. *Biometrics* **11**, 375–386 (1955)
22. Loh, W.-Y., Eltinge, J., Cho, M.J., Li, Y.: Classification and regression trees and forests for incomplete data from sample surveys. *Statistica Sinica* **29**, 431–453 (2019)
23. Breiman, L.: Random forests. *Machine Learning* **45**(1), 5–32 (2001)
24. Wilson, E.B., Hilferty, M.M.: The distribution of chi-square. *Proc. Natl. Acad. Sci. USA* **17**, 684–688 (1931)
25. Zeileis, A., Hothorn, T., Hornik, K.: Model-based recursive partitioning. *J. Comput. Graph. Stat.* **17**, 492–514 (2008)
26. Loh, W.-Y., Zhang, Q., Zhang, W., Zhou, P.: Missing data, imputation and regression trees. *Statistica Sinica* **30**, 1697–1722 (2020)
27. NHTSA: Test Reference Guide Version 5, <https://one.nhtsa.gov/Research/Databases-and-Software/NHTSA-Test-Reference-Guides>, **1** (2014)



Wei-Yin Loh is Professor of Statistics at the University of Wisconsin, Madison. He has a PhD from the University of California, Berkeley, and is a fellow of the American Statistical Association and the Institute of Mathematical Statistics. He has been studying classification and regression tree algorithms for more than 35 years and is the author of the GUIDE algorithm and software.



Detecting Outliers and Influential and Sensitive Observations in Linear Regression

31

Daniel Peña

Contents

31.1	Introduction	605
31.2	Residuals and Leverage in the Regression Model ...	607
31.3	Diagnosis for a Single Outlier	608
31.3.1	Outliers.....	608
31.3.2	Influential Observations.....	608
31.3.3	The Relationship Between Outliers and Influential Observations.....	609
31.4	Diagnosis for Groups of Outliers	610
31.4.1	Methods Based on an Initial Clean Set.....	610
31.4.2	Methods Based on Eigenvectors of the Influence and Sensitivity Matrix.....	611
31.5	Robust Regression	613
31.6	Detecting Outliers in Large Data Sets	614
31.7	Examples	615
31.7.1	A Simulated Example.....	615
31.7.2	The Housing Boston Data.....	616
31.8	Final Remarks	617
	References	618

Abstract

This chapter reviews diagnostic and robust procedures for detecting outliers and other interesting observations in linear regression. First, we present statistics for detecting single outliers and influential observations and show their limitations for multiple outliers in high-leverage situations. Second, we discuss diagnostic procedures designed to avoid masking by finding first a clean subset for estimating the parameters and then increasing its size by

incorporating, one by one, new homogeneous observations until a heterogeneous observation is found. We also discuss procedures based on sensitive observations for detecting high-leverage outliers in large data sets using the eigenvectors of a sensitivity matrix. We briefly review robust estimation methods and its relationship with diagnostic procedures. Next, we consider large high-dimensional data sets where the application of iterative procedures can be slow and show that the joint use of simple univariate statistics, as predictive residuals, Cook’s distances, and Peña’s sensitivity statistic, can be a useful diagnostic tool. We also comment on other recent procedures based on regularization and sparse estimation and conclude with a brief analysis of the relationship of outlier detection and cluster analysis. A real data and a simulated example are presented to illustrate the procedures presented in the chapter.

Keywords

Cook’s distance · Influential observations · Influence matrix · Leverage · Masking · Predictive residuals · Sensitivity matrix · Robust estimation

31.1 Introduction

Data often contains outliers or atypical observations. Outliers are points which are heterogeneous with the rest, due to large measurement errors, different experimental conditions, or unexpected variability. Detecting these observations is important: first, because they can lead to new discoveries and, second, because they can modify completely the conclusions we draw from the data. For instance, penicillin was found because Pasteur instead of ignoring an outlier tried to understand the reason of an atypical effect. As Box [1] has emphasized: “Every operating system supplies information

D. Peña (✉)
 Department of Statistics, Universidad Carlos III de Madrid, Madrid, Spain
 e-mail: daniel.pena@uc3m.es

on how it can be improved and if we use this information it can be a source of continuous improvement.” A way in which this relevant information may appear is in outlying observations that, in many engineering processes, can show the way to improve the process. For instance, in a production process, a large value in one of the variables we monitor may be due, among other causes, to (1) a large value of one of the input control variables, (2) an unexpected interaction among the input variables, and (3) a large measurement error due to some defect in the measurement instrument. In the first case, the outlying observations may provide no new information about the performance of the process; in the second case, may lead to a potentially useful discovery; and in the third one, to an improvement of the process control.

Detection of outliers has a long tradition in engineering statistics, see Chapter 1 of Hampel et al. [2], and now it has a growing importance in the big data environment. Many communications and controlling devices are automatically collecting large data sets with many observations and variables using wireless sensor networks. Sensor nodes sometime fail to record the data correctly, due to depletion of batteries or environmental influence, and congestion in communication may lead to packet loss. These failures will produce outliers, and their identification is very important when building regression models as otherwise we may fail to explain the relationship between the variables in the data set. An alternative way to deal with the problem of contaminated data is to use robust methods that avoid large outliers effects on the estimation of the parameters, and this approach will be also briefly discussed.

This chapter defines outliers, influential and sensitive observations in regression models, and presents diagnostic methods to detect them. Outliers are observations that are very different from its forecast, or interpolation, with the rest of the available data. Note that this implies that an outlier is defined with respect to a model, or in general a rule to forecast the observations, and, therefore, an observation can be an outlier given a model and a common point under another. Influential observations are those which have a strong influence on the estimation of the model parameters. Again, as outliers, they are model dependent. These observations are obtained by looking at the standardized change they produce in the parameter vector or in the vector of forecasts. Influence is a global analysis. A third type of interesting points in the sample is sensitive observations which are those points that suffer an unusual influence from the rest of the sample. Sensitivity is more a local concept, it shows how each observation is affected by the rest, and it is computed by deleting each sample point in turn and looking at the change that these modifications produce in the forecast of a single point. We will see that influence and sensitivity are important concepts for understanding the effect of data in building a regression model and in finding groups of outliers.

Many procedures are available to identify a single outlier or an isolated influential point in linear regression. The books of Belsley et al. [3], Hawkins [4], Cook and Weisberg [5], Atkinson [6], Chatterjee and Hadi [7], Barnett and Lewis [8], Atkinson and Riani [9], and Carroll [10] present good analyses of this problem. To identify outliers and to measure influence, each point can be deleted, as proposed by Cook [11] and Belsley et al. [3], or its weight decreased, as in the local influence analysis introduced by Cook [12]. See Suárez Rancel and González Sierra [13] and Hartless et al. [14] for a review of local influence in regression and many references. A related way to analyze influence has been proposed by Critchley et al. [15] by an extension of the influence-curve methodology.

The detection of influential subsets, or multiple outliers, is more difficult due to the masking and swamping problems. Masking occurs when one outlier is not detected because of the presence of others; swamping happens when a non-outlier is wrongly identified due to the effect of some hidden outliers, see Lawrance [16]. Several procedures have been proposed for dealing with multiple outliers; see Hawkins et al. [17], Gray and Ling [18], Marasinghe [19], Kianifard and Swallow [20,21], Hadi and Simonoff [22,23], Atkinson [24], and Swallow and Kianifard [25]. A different analysis for detecting groups of outliers by looking at the eigenvectors of an influence matrix, or even better those of a sensitivity matrix, is presented by Peña and Yohai [26,27].

We briefly discuss in this chapter the relationship between these diagnostic procedures and robust regression methods. See Huber [28] and Hampel et al. [2] for good discussions of the complementary role of diagnosis and robustness and for robust estimation in regression and the books by Rousseeuw and Leroy [29] and Maronna et al. [30] for robust regression estimates. This problem has also received attention in the Bayesian literature since the seminal article of Box and Tiao [31]. See Peña and Guttman [32], Berger et al. [33], Justel and Peña [34], and Hans [35] for a Bayesian approach to this problem and references.

The chapter is organized as follows. In Sect. 31.2 we present the regression model and the notation we will use in this chapter and define the main measures which will be introduced for outlier analysis. Section 31.3 reviews procedures for detecting single outliers and influential observations in regression. In Sect. 31.4 we discuss the multiple outlier problem and two types of diagnostic procedures: first, those based on an initial clean subset and, second, those based on eigenvalue analysis of some diagnostic matrices. Section 31.5 briefly discusses robust estimation and its connection to diagnostic methods. Section 31.6 presents some fast procedures to find outliers in large data sets. In particular we introduce a simple statistic proposed by Peña [36] as a diagnostic tool for large data set that avoids the masking problem. Section 31.7 includes two ex-

amples, one simulated and the other with real data, to illustrate some of the diagnostic methods presented for detecting groups of outliers. Section 31.8 summarizes some concluding remarks.

31.2 Residuals and Leverage in the Regression Model

We assume that we have observed a sample of size n of a random variable $\mathbf{y} = (y_1, \dots, y_n)'$ and a set of $p - 1$ explanatory variables which are linearly related by

$$y_i = \mathbf{x}_i' \boldsymbol{\beta} + u_i, \quad (31.1)$$

where the u_i are the measurement errors, which will be independent normal zero mean random variables with variance σ^2 and $\mathbf{u} = (u_1, \dots, u_n)'$. The $\mathbf{x}_i = (1, x_{2i}, \dots, x_{pi})'$ s are numerical vectors in R^p , and we will denote by \mathbf{X} the $n \times p$ matrix of rank p whose i -th row is \mathbf{x}_i' . The usual criterion to estimate the parameters $\boldsymbol{\beta}$ is to minimize

$$\sum_{i=1}^n (y_i - \mathbf{x}_i' \boldsymbol{\beta})^2 \quad (31.2)$$

that is the least squares (LS) criterion that is optimal for normally distributed data. The LS estimate of $\boldsymbol{\beta}$ is obtained by projecting the vector \mathbf{y} on the space generated by the columns of \mathbf{X} , which leads to

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y},$$

and the vector of fitted values, $\hat{\mathbf{y}} = (\hat{y}_1, \dots, \hat{y}_n)'$, is given by

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{H}\mathbf{y}, \quad (31.3)$$

where $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ is the idempotent projection matrix. The vector orthogonal to the space generated by the \mathbf{X} variables is the residual vector, $\mathbf{e} = (e_1, \dots, e_n)'$, which is defined by

$$\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}} = (\mathbf{I} - \mathbf{H})\mathbf{y}, \quad (31.4)$$

and we will let $\hat{\sigma}_R^2 = \mathbf{e}'\mathbf{e}/(n - p)$ be the estimated residual variance.

From (31.4), inserting $\mathbf{X}\boldsymbol{\beta} + \mathbf{u}$ instead of \mathbf{y} and using $\mathbf{H}\mathbf{X} = \mathbf{X}$, we obtain the relationship between the residuals and the measurement errors, $\mathbf{e} = (\mathbf{I} - \mathbf{H})\mathbf{u}$. Thus, each residual is a linear combination of the measurement errors. Letting $h_{ij} = \mathbf{x}_i'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_j$ be the elements of the matrix, \mathbf{H} , we have

$$e_i = u_i - \sum_{j=1}^n h_{ij}u_j \quad (31.5)$$

and, if the second term is small, the residual e_i will be close to the measurement error, u_i . The variance of this second term is

$$\text{var}\left(\sum_{j=1}^n h_{ij}u_j\right) = \sigma^2 \sum_{j=1}^n h_{ij}^2 = \sigma^2 h_{ii}$$

and if h_{ii} , the diagonal term of \mathbf{H} , is large, the difference between the residual and the measurement error can be large. The values h_{ii} are called the leverage of the observation and measure the discrepancy of each observation \mathbf{x}_i with respect to the mean of the explanatory variables. It can be shown (see, for instance, [11] p. 12) that

$$h_{ii} = \mathbf{x}_i'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i = \frac{1}{n} \left(1 + (\tilde{\mathbf{x}}_i - \bar{\mathbf{x}})' \mathbf{S}_{xx}^{-1} (\tilde{\mathbf{x}}_i - \bar{\mathbf{x}})\right)$$

where $\tilde{\mathbf{x}}_i = (x_{2i}, \dots, x_{pi})$ does not include the constant term, $\bar{\mathbf{x}}$ is the vector of means of the $p - 1$ explanatory variables, and \mathbf{S}_{xx} is their covariance matrix. Note that if the explanatory variables were uncorrelated, h_{ii} would be the sum of the standardized distances to the means, $((x_{ij} - \bar{x}_j)/s_j)^2$. As $\sum_{i=1}^n h_{ii} = \text{tr}(\mathbf{H}) = p$, the average value of the leverage is $\bar{h} = \sum h_{ii}/n = p/n$, and it can be shown that $1/n \leq h_{ii} \leq 1$. From (31.5) we conclude that the residual will be close to the measurement error for those observations close to the center of the explanatory data, where $h_{ii} \simeq 1/n$, but will be very different for the extreme points where $h_{ii} \simeq 1$. The residual covariance matrix is

$$\text{Var}(\mathbf{e}) = E[\mathbf{e}\mathbf{e}'] = E((\mathbf{I} - \mathbf{H})\mathbf{u}\mathbf{u}'(\mathbf{I} - \mathbf{H})) = \sigma^2(\mathbf{I} - \mathbf{H}) \quad (31.6)$$

and $\text{Var}(e_i) = \sigma^2(1 - h_{ii})$, which will be large when $h_{ii} \simeq 1/n$ and close to zero if $h_{ii} \simeq 1$. As the mean of the residuals is zero if the variance of e_i is very small, this implies that its value will be close to zero, whatever the value of u_i is.

To avoid residuals with different variances, the standardized residuals are defined as

$$r_i = \frac{e_i}{\hat{\sigma}_R \sqrt{1 - h_{ii}}} \quad (31.7)$$

which will have variance equal to one. A third type of useful residuals are the predictive, deleted, or out-of-sample residuals, defined by $e_{(i)} = y_i - \hat{y}_{i(i)}$, where $\hat{y}_{i(i)}$ is computed in a sample with the i th observation deleted. It can be shown that

$$e_{(i)} = \frac{e_i}{(1 - h_{ii})} \quad (31.8)$$

and the variance of these predictive residuals is $\sigma^2/(1 - h_{ii})$. If we estimate σ^2 by $\hat{\sigma}_{(i)R}^2$, the residual variance in a regression which does not include the i th observation, the standardization of the predictive residual leads to the studentized residual, defined by

$$\hat{t}_i = \frac{e_i}{\hat{s}_{R(i)}\sqrt{1-h_{ii}}} \quad (31.9)$$

which has a student t distribution with $n-p-1$ degrees of freedom. An alternative useful expression of these residuals is based on $h_{ii(i)} = \mathbf{x}'_i (\mathbf{X}_{(i)}\mathbf{X}_{(i)})^{-1} \mathbf{x}_i = h_{ii}/(1-h_{ii})$, where $\mathbf{X}_{(i)}$ is the $(n-1) \times p$ matrix without the row \mathbf{x}'_i , and therefore, we have the alternative expression

$$\hat{t}_i = \frac{e^{(i)}}{\hat{s}_{R(i)}\sqrt{1+h_{ii(i)}}}. \quad (31.10)$$

31.3 Diagnosis for a Single Outlier

31.3.1 Outliers

If one observation, y_h , does not follow the regression model, either because its expected value is not $\mathbf{x}'_h\boldsymbol{\beta}$ or its conditional variance is not σ^2 , we will say that this observation is an outlier. The discrepancy is usually translated to the residual. For instance, if the h th observation has not been generated by the linear model $y_h = \mathbf{x}'_h\boldsymbol{\beta} + u_h$, but by a different model, $g(\mathbf{x}'_h) + u_h$, then

$$e_h = g(\mathbf{x}'_h) - \mathbf{x}'_h\hat{\boldsymbol{\beta}} + u_h$$

and the h th point will be an outlier if the deviation $|g(\mathbf{x}'_h) - \mathbf{x}'_h\hat{\boldsymbol{\beta}}|$ is much larger than $|\mathbf{x}'_h(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})|$. Suppose, in order to simplify, that $g(\mathbf{x}'_h) = \mathbf{x}'_h\boldsymbol{\alpha}$, that is, the data are also generated by a linear model but with different parameter values. Then, even if $\boldsymbol{\alpha}$ is very different from $\boldsymbol{\beta}$, the size of $|\mathbf{x}'_h(\boldsymbol{\alpha} - \hat{\boldsymbol{\beta}})|$ depends on \mathbf{x}'_h , and the discrepancy between the parameter values would be easier to detect when $|\mathbf{x}'_h|$ is large than when it is small. When the observation is an outlier because its measurement error comes from a different distribution, for instance, one with variance $k\sigma^2$, instead of σ^2 , where $k > 1$, we expect that $|u_h|$ will be larger than the rest of the measurement errors. It is intuitive, and it has been formally shown [33] that we cannot differentiate between a change in the mean and a change in the variance of a distribution by using just one observation; also models which assume a change in the variance are equivalent to those which assume shifts in the mean of the observations. Thus, a simple model for a single outlier at time h th is to consider the mean shift model

$$y_h = \mathbf{x}'_h\boldsymbol{\beta} + w + u_h \quad (31.11)$$

where w is the size of the outlier and u_h is $N(0, \sigma^2)$. In general, the size of a possible outlier at every point is estimated by the parameter w in the model

$$y_i = \mathbf{x}'_i\boldsymbol{\alpha} + wI_i^{(h)} + u_i, \quad i = 1, \dots, n \quad (31.12)$$

where $I_i^{(h)}$ is a dummy variable given by $I_i^{(h)} = 1$, when $i = h$, and $I_i^{(h)} = 0$ otherwise. We can test for outliers by fitting this model for $h = 1, \dots, n$ and checking if the estimated coefficient \hat{w}_h is significant. It is easy to show that:

- (1) $\hat{\boldsymbol{\alpha}}_h = (\mathbf{X}'_{(h)}\mathbf{X}_{(h)})^{-1} \mathbf{X}'_{(h)}\mathbf{y}_{(h)} = \hat{\boldsymbol{\beta}}_{(h)}$, the regression parameters in the model with the dummy variable at point h th are estimated by deleting the case (y_h, \mathbf{x}_h) affected by the dummy variable and applying to the rest of the observations the usual procedure;
- (2) $\hat{w}_h = y_h - \mathbf{x}'_h\hat{\boldsymbol{\alpha}}_h$, and, therefore, the estimated residual at the h th point is zero, as $e_h = y_h - \mathbf{x}'_h\hat{\boldsymbol{\alpha}}_h - \hat{w}_h = 0$.
- (3) The t statistic to check if the parameter \hat{w}_h is significant is equal to the studentized residual, t_h , as defined in (31.9).

Assuming that only one observation is an outlier, the test is made by comparing the standardized residual to the maximum of a t distribution with $n-p-2$ degrees of freedom. Often, for moderate n , a point is considered as an outlier if its studentized residual is larger than 3.5.

31.3.2 Influential Observations

An intuitive way to measure the effect of an observation on the vector of estimated parameters, or in the vector of forecasts, is to delete this observation from the sample and see how its deletion affects these vectors. A measure of the influence of the i th observation on the parameter estimate is given by

$$D(i) = \frac{(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)})\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)})}{p\hat{s}_{R}^2}, \quad (31.13)$$

which, as the covariance of $\hat{\boldsymbol{\beta}}$ is $\hat{s}_{R}^2(\mathbf{X}'\mathbf{X})^{-1}$, measures the change between $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\beta}}_{(i)}$ with relation to the covariance matrix of $\hat{\boldsymbol{\beta}}$, standardized by the dimension of the vector p . This measure was introduced by Cook [11]. Of course, other standardizations are possible, Belsley et al. [3] proposed using instead of \hat{s}_{R}^2 , the variance of the regression model when the i th observation is deleted, $\hat{s}_{(i)R}^2$, and Diaz-García and Gonzalez-Farias [37] have suggested standardizing the vector $(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)})$ by its variance, instead of using the variance of $\hat{\boldsymbol{\beta}}$. See Cook, Peña and Weisberg [38] for a comparison of some of these possible standardizations.

Equation (31.13) can also be written as the standardized change in the vector of forecast:

$$D_i = \frac{(\hat{\mathbf{y}} - \hat{\mathbf{y}}_{(i)})' (\hat{\mathbf{y}} - \hat{\mathbf{y}}_{(i)})}{p \hat{s}_R^2} \tag{31.14}$$

where $\hat{\mathbf{y}}_{(i)} = \mathbf{X} \hat{\boldsymbol{\beta}}_{(i)} = (\hat{y}_{1(i)}, \dots, \hat{y}_{n(i)})'$. Note that from (31.3) we have that $\text{var}(\hat{y}_i) = \sigma^2 h_{ii}$ and as the average value of h_{ii} is p/n , (31.14) is standardized by this average value and by the dimension, n , of the vector. A third way to measure the influence of the i th point is to compare \hat{y}_i with $\hat{y}_{(i)}$, where $\hat{y}_{(i)} = \mathbf{x}_i' \hat{\boldsymbol{\beta}}_{(i)}$. With the usual standardization by the variance, we have

$$D_i = \frac{(\hat{y}_i - \hat{y}_{(i)})^2}{p \hat{s}_R^2 h_{ii}} \tag{31.15}$$

and using the relation between the inverse of $\mathbf{X}'\mathbf{X}$ and $\mathbf{X}'_{(i)}\mathbf{X}_{(i)}$, we obtain

$$\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}_{(i)} = (\mathbf{X}'\mathbf{X})^{-1} \mathbf{x}_i \frac{e_i}{1 - h_{ii}} \tag{31.16}$$

inserting this in (31.13), it is easy to see that (31.15) is equivalent to (31.13) and (31.14). Also, as from (31.16) we have that

$$\hat{\mathbf{y}} - \hat{\mathbf{y}}_{(i)} = \mathbf{h}_i \frac{e_i}{1 - h_{ii}} \tag{31.17}$$

where \mathbf{h}_i is the i th column of the \mathbf{H} matrix. Using this expression in (31.14), we obtain a convenient way to compute Cook's statistic:

$$D_i = \frac{r_i^2 h_{ii}}{p(1 - h_{ii})} \tag{31.18}$$

where r_i is the standardized residual given by (31.7). For large n , the expected value of D_i can be approximated by

$$E(D_i) \simeq \frac{h_{ii}}{p(1 - h_{ii})}, \tag{31.19}$$

and it will be very different for observations with different leverage.

Cook proposed judging the values of D_i by an $F(p; n - p; 1 - \alpha)$, where F is the distribution used in building a confidence region for the $\boldsymbol{\beta}$ parameters. Thus, we may identify points as influential when they are able to move the estimate out of the confidence region for a fixed value of α and declare as influential those observations which verify $D_i \geq F(p; n - p; 1 - \alpha)$. This solution is not satisfactory for large sample size because it is difficult that any observation is deemed as influential. Muller and Mok [39] have obtained the distribution of the D_i for normal explanatory variables, but this distribution is complicated.

Cook [12] proposed a procedure for the assessment of the influence on a vector of parameters $\boldsymbol{\theta}$ of minor perturbations of a statistical model. This approach is very flexible and can be used to see the effect of small perturbations which would not normally be detected by deletion of one observation. He suggested introducing a $n \times p$ vector $\boldsymbol{\omega}$ of case weights and

using the likelihood displacement $(L(\hat{\boldsymbol{\theta}}) - L(\hat{\boldsymbol{\theta}}_\omega))$, where $\hat{\boldsymbol{\theta}}$ is the maximum likelihood (ML) estimator of $\boldsymbol{\theta}$ and $\hat{\boldsymbol{\theta}}_\omega$ the ML when the case weight ω is introduced. Then, he showed that the directions of greatest local change in the likelihood displacement for the linear regression model are given by the eigenvectors linked to the largest eigenvalues of the curvature matrix, $\mathbf{L} = \mathbf{EHE}$, where \mathbf{E} is the vector of residuals. Later, we will see the relation of this approach to some procedures for multiple outlier detection.

31.3.3 The Relationship Between Outliers and Influential Observations

An outlier may or may not be an influential observation, and an influential observation may or may not be an outlier. To illustrate this point, consider the data in Table 31.1. We will use these data to build four data sets. The first one includes cases 1 to 9 repeated three times and has sample size $n = 27$. The other three sets are formed by adding a new observation to this data set. The set (a) is built by adding case 28(a), the set (b) by adding case 28(b), and the set (c) by adding case 28(c). Table 31.2 shows some statistics of these four data sets where (0) refers to the set of 27 observations and (a), (b), and (c) to the sets of 28 observations, as defined before. The table gives the values of the estimated parameters, their t statistics under parenthesis, the residual standard deviation, the leverage of the added point, the standardized residual for the added point, and the value of Cook's statistics.

In set (a) observation 28 is clearly an outlier with a value of the standardized residual of 4.68, but it is not influential, as $D_{28(a)} = 0.92$ which is a small value. In case (b) the 28th point is not an outlier as $r_{28(b)} = 1.77$ is not significant, but

Table 31.1 Three sets of data which differ in one observation

Case	1	2	3	4	5	6	7	8	9	(a)	(b)	(c)
\mathbf{x}_1	-2	0	2	-4	3	1	-3	-1	4	0	-3	-3
\mathbf{x}_2	6, 5	7, 3	8, 3	6, 0	8, 8	8, 0	5, 9	6, 9	9, 5	7, 2	9, 7	7, 3
\mathbf{y}	-1, 5	0, 5	1, 6	-3, 9	3, 5	0, 8	-2, 7	-1, 3	4, 1	5	-1, 5	4

Table 31.2 Some statistics for the three regression fitted to the data in Table 31.1

	$\hat{\beta}_0$	$\hat{\beta}_2$	$\hat{\beta}_1$	\hat{s}_R	h_{28}	r_{28}	D_{28}
(0)	2.38 (0.82)	-0.30 (0.78)	1.12 (6.24)	0.348	-	-	-
(a)	13.1 (1.7)	-1.72 (-1.66)	1.77 (3.69)	0.96	0.11	4.68	0.92
(b)	-2.74 (-2.9)	0.38 (3.08)	0.80 (13.87)	0.36	0.91	1.77	11.1
(c)	-25.4 (-5.41)	3.43 (5.49)	-0.624 (2.22)	0.91	0.65	4.63	13.5

it is very influential as is indicated by the large D_{28} value. Finally, in set (c) the observation is both an outlier, $r_{28} = 4.63$, and very influential, $D_{28} = 13.5$.

Note that if the leverage is small $h_{ii} \simeq 1/n$, $h_{ii}/(1-h_{ii}) \simeq (n-1)^{-1}$, and by (31.18)

$$D_i = \frac{r_i^2}{p} \left(\frac{1}{n-1} \right)$$

then if n is large, the observation cannot be influential, whatever the value of r_i^2 . On the other hand, high-leverage observations with h_{ii} close to one will have a ratio $h_{ii}/(1-h_{ii})$ arbitrarily large and, even if r_i^2 is small, will be influential

31.4 Diagnosis for Groups of Outliers

The procedures that we have presented in the previous section are designed for a single outlier. We can extend these ideas to multiple outliers as follows. Let I be an index set corresponding to a subset of r data points. The checking of this subset can be done by introducing dummy variables, as in the univariate case. Assuming normality, the F test for the hypothesis that the coefficients of the dummy variables are zero is given by

$$F_{r,(n-p-r)} = \frac{\mathbf{e}'_I(\mathbf{I} - \mathbf{H}_I)^{-1}\mathbf{e}_I}{r\hat{s}_{(r)R}^2}$$

where \mathbf{e}_I is the vector of least squares residuals; \mathbf{H}_I the $r \times r$ submatrix of \mathbf{H} , corresponding to the set of observations included in I ; and $\hat{s}_{(r)R}^2$ the residual variance of the regression with the set I deleted. Cook and Weisberg [5] proposed to measure the joint influence of the data points with index in I by deleting the set I and computing, as in the single outlier case,

$$D_I = \frac{(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(I)})\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(I)})}{p\hat{s}_R^2}$$

which can also be written, as a generalization of (31.18), by $D_I = (\mathbf{e}'_I(\mathbf{I} - \mathbf{H}_I)^{-1}\mathbf{H}_I(\mathbf{I} - \mathbf{H}_I)^{-1}\mathbf{e}_I)/p\hat{s}_R^2$. Note that a large value of D_I may be due to a single influential observation included in the set I , but it can also be due to the sum of small individual effects of a set of observations that are masking each other. However, in the first case, this single observation will be easily identified. Also, a subset of individually highly influential points, whose effect is to cancel each other out, will lead to a small value of D_I , but, again in this case, the individual effects will be easy to identify. However, to build this measure, we should compute all the set of I in the n data, and this task will be impossible to do for large I and n .

The procedures for finding multiple outliers in regression can be divided into three main groups. The first one is based on robust estimation that will be discussed in Sect. 31.5. The

main idea is to compute an estimate that is not affected by the outliers and then find the outliers as those cases with large residuals with respect to the robust fit. A second class of procedures uses also robust ideas to build an initial clean subset and then combine least squares estimates in clean subsets and diagnosis ideas for outlier detection. Three procedures in this spirit will be presented next, and they can be very effective when p and n are not large. For large data sets with many predictors and high-leverage observations, robust estimates can be very difficult to compute, and procedures based on the clean set idea may not work well, because of the difficulty in selecting the initial subset. The third type of procedures is based on the eigen-structure analysis of some diagnostic matrices and is specially useful for large data sets.

31.4.1 Methods Based on an Initial Clean Set

Kianifard and Swallow [20, 21] proposed to build a “clean” set of observations and check the rest of the data with respect to this set. If the observation closest to the clean set is not an outlier, then increase the “clean” set by one observation, and continue checking points until no new observation can be incorporated into the basic set. The key step in this procedure is to find the initial subset because if it contains outliers, all the procedure breaks down. These authors proposed using either the predictive or standardized residuals or a measure of influence as D_i .

A similar procedure was proposed by Hadi and Simonoff [22, 23]. They recommend building the initial subset using as robust estimate the least median of squares estimate (LMS) that will be presented in Sect. 31.5. The clean set is built by computing this robust estimate and then uses the $h = \lceil \frac{n+p+1}{2} \rceil$ observations with the smallest residuals with respect to this robust fit to form the initial “clean” set, which we call M . The procedure continues by fitting a regression model by least squares to this clean set, M . Calling $\hat{\boldsymbol{\beta}}_M$ the LS estimated parameters and $\hat{\sigma}_M$ the residual standard deviation, a set of in-sample and out-of-sample residuals is obtained as follows:

$$d_i = \frac{|y_i - \mathbf{x}'_i\hat{\boldsymbol{\beta}}_M|}{\hat{\sigma}_M\sqrt{1 - \mathbf{x}'_i(\mathbf{X}'_M\mathbf{X}_M)^{-1}\mathbf{x}_i}}, \quad \text{if } i \in M$$

$$d_i = \frac{|y_i - \mathbf{x}'_i\hat{\boldsymbol{\beta}}_M|}{\hat{\sigma}_M\sqrt{1 + \mathbf{x}'_i(\mathbf{X}'_M\mathbf{X}_M)^{-1}\mathbf{x}_i}}, \quad \text{if } i \notin M$$

That is, d_i represents the standardized residual (31.7) for the data in set M and the predictive residual (31.10) for observations outside this set. Then, all the observations are arranged in increasing order according to d_i . Let s be the size of the set M (which is h in the first iteration, but will be changing as explained below). If $d_{(s+1)}$ is smaller than some critical value, a new set of size $s+1$ is built with the

$s + 1$ observations with smallest d values. If $d_{(s+1)}$ is larger than some critical value, all observations out of the set M are declared as outliers, and the procedure stops. If $n = s + 1$, we stop and declare no outliers in the data. These authors proposed using as critical values those of the t distribution adjusted by Bonferroni, that is, $t(\frac{\alpha}{2(s+1)}, s - p)$.

Atkinson [24, 40] proposed a similar approach called the forward search. His idea is again to combine a robust estimate with diagnostic analysis. He computes also the LMS estimate but instead of generating a large set of candidates by random sample, he generates a set of candidate values of $\hat{\boldsymbol{\beta}}$ fitting by least squares subsamples of size $p, p + 1, \dots, n$. The procedure is as follows. We start generating a random sample of size p , let I_p the indices of the observations selected. Then, we compute the parameters $\hat{\boldsymbol{\beta}}(p)$ by LE and the residual for all the cases, $\mathbf{e} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}(p)$. The residuals are corrected by

$$\begin{aligned} u_i^2 &= e_i^2, & i \in I \\ u_i^2 &= e_i^2 / (1 + h_{ii}), & i \notin I \end{aligned} \quad (31.20)$$

and these residuals, u_i^2 , are ordered and the smallest $p + 1$ are selected. With this new sample of size $m = p + 1$, the process is repeated, that is, the parameters are computed by LE and the residuals to this fit for the n points are obtained. The corrected residuals (31.20) are computed and the process is continued. In this way we obtain a set of estimates, $\hat{\boldsymbol{\beta}}(m)$, $m = p, \dots, n$, the corresponding residuals; $\mathbf{e}(m) = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}(m)$; and the robust scales (31.27), $s(\boldsymbol{\beta}(m))$. The value selected is the $\hat{\boldsymbol{\beta}}(m)$ which minimizes the robust scale. This process is a complete forward search, and several forward searches are done starting by different random samples. The residuals are then identified by using this LMS estimate

computed from several forward searches. An improvement of this procedure was proposed by Atkinson and Riani [9], which separates clearly the estimation of the clean subset and the forward search. The initial estimate is computed as proposed by LMS, by taking many random samples of size p . Then, the forward search is applied but stressing the use of diagnostic statistics for monitoring the performance of the procedure. Finally, Swallow and Kianifard [25] suggest also a similar procedure which uses a robust estimate of the scale and determines the cutoff values for the testing from simulations.

These procedures work when both p and n are not large and the proportion of outliers is moderate, as shown in the simulated comparison by Wisnowski et al. [41]. However, they do not work as well in large data sets with high contamination. The LMS estimates rely on having at least a sample of size p without outliers, and we need an unfeasible number of samples to have a large probability of this event when p and n are large (see [27]). This good initial estimate is the key for the procedures of the clean set. In the next section, we will present more effective procedures for large data sets.

31.4.2 Methods Based on Eigenvectors of the Influence and Sensitivity Matrix

The matrix of forecast changes was defined by Peña and Yohai [26] by looking at the vectors of changes in the forecast vector when each observation is deleted. Calling as before \hat{y}_i to the forecast of the i th observation and $\hat{y}_{i(j)}$ to the forecast when the parameters are computed deleting the j th observations, this matrix of forecast changes is given by

$$\mathbf{T} = \begin{bmatrix} \hat{y}_1 - \hat{y}_{1(1)} & \hat{y}_1 - \hat{y}_{1(2)} & \cdots & \hat{y}_1 - \hat{y}_{1(n-1)} & \hat{y}_1 - \hat{y}_{1(n)} \\ \hat{y}_2 - \hat{y}_{2(1)} & \hat{y}_2 - \hat{y}_{2(2)} & \cdots & \hat{y}_2 - \hat{y}_{2(n-1)} & \hat{y}_2 - \hat{y}_{2(n)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \hat{y}_{n-1} - \hat{y}_{n-1(1)} & \hat{y}_{n-1} - \hat{y}_{n-1(2)} & \cdots & \hat{y}_{n-1} - \hat{y}_{n-1(n-1)} & \hat{y}_{n-1} - \hat{y}_{n-1(n)} \\ \hat{y}_n - \hat{y}_{n(1)} & \hat{y}_n - \hat{y}_{n(2)} & \cdots & \hat{y}_n - \hat{y}_{n(n-1)} & \hat{y}_n - \hat{y}_{n(n)} \end{bmatrix}$$

The columns of this matrix are the vectors $\mathbf{t}_i = \hat{\mathbf{y}} - \hat{\mathbf{y}}_{(i)}$, and the Cook's statistic (31.14) is their standardized norm. These vectors can also be written as $\mathbf{t}_i = \mathbf{e}_{(i)} - \mathbf{e}$, where $\mathbf{e}_{(i)}$ is the vector of residuals when observation i th is deleted. Therefore, \mathbf{T} can also be considered the matrix of residual changes. Peña and Yohai [26] define the $n \times n$ influence matrix \mathbf{M} as

$$\mathbf{M} = \frac{1}{ps_R^2} \mathbf{T}'\mathbf{T}.$$

The matrix \mathbf{T} has columns $\mathbf{X}(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)})$, and using (31.16) it is easy to see that the ij -th element of \mathbf{M} is

$$m_{ij} = \frac{e_i e_j h_{ij}}{(1 - h_{ii})(1 - h_{jj})ps_R^2} = \frac{e_{i(i)} e_{j(j)} h_{ij}}{ps_R^2}.$$

and \mathbf{M} can be written as

$$\mathbf{M} = \frac{1}{ps_R^2} \mathbf{W}\mathbf{H}\mathbf{W} \quad (31.21)$$

where \mathbf{W} is the diagonal matrix of predictive residuals (31.8). Assuming that all the residuals are different from zero, from (4) the rank of \mathbf{M} is equal to p , the rank of \mathbf{H} . Observe that the diagonal elements of \mathbf{M} are the Cook's statistics.

Let $r_{ij} = m_{ij}/m_{ii}^{1/2}m_{jj}^{1/2}$ be the uncentered correlation coefficient between \mathbf{t}_i and \mathbf{t}_j . Let us show that the eigenvectors of the matrix \mathbf{M} will be able to indicate groups of influential observations. Suppose that there are k groups of influential observations I_1, \dots, I_k , such that:

- (i) If $i, j \in I_h$, then $|r_{ij}| = 1$. This means that the effects on the least squares fit produced by the deletion of two points in the same set I_h have correlation 1 or -1 .
- (ii) If $i \in I_j$ and $l \in I_h$ with $j \neq h$, then $r_{il} = 0$. This means that the effects produced on the least squares fit by observations i and j belonging to different sets are uncorrelated.
- (iii) If i does not belong to any I_h , then $m_{ij} = 0$ for all j . This means that data points outside these groups have no influence on the fit.

Now, according to (i) we can split each set I_h in I_h^1 and I_h^2 such that: (1) If $i, j \in I_h^q$, then $r_{ij} = 1$. (2) If $i \in I_h^1$ and $j \in I_h^2$, then $r_{ij} = -1$. Let $\mathbf{v}_1 = (v_{11}, \dots, v_{1n})', \dots, \mathbf{v}_k = (v_{k1}, \dots, v_{kn})'$ be defined by $v_{hj} = m_{ij}^{1/2}$ if $j \in I_h^1$, $v_{hj} = -m_{ij}^{1/2}$ if $j \in I_h^2$, and $v_{hj} = 0$ if $j \notin I_h$. Then, if (i)–(iii) hold, by (6) the matrix \mathbf{M} is

$$\mathbf{M} = \sum_{i=1}^k \mathbf{v}_i \mathbf{v}_i'$$

and since the \mathbf{v}_i 's are orthogonal, the eigenvectors of \mathbf{M} are $\mathbf{v}_1, \dots, \mathbf{v}_k$, and the corresponding eigenvalues $\lambda_1, \dots, \lambda_k$ are given by

$$\lambda_h = \sum_{i \in I_h} m_{ii}.$$

It is clear that when the matrix \mathbf{M} satisfies (i)–(iii), the only sets I with large C_I are I_h^q , $1 \leq h \leq k$, $q = 1, 2$, and these sets may be found by looking at the eigenvectors associated with non-null eigenvalues of \mathbf{M} . Note that (6) can also be written as

$$r_{ij} = \text{sign}(e_i)\text{sign}(e_j)h_{ij}/(h_{ii}h_{jj})^{1/2}$$

which means that, in the extreme case that we have presented, the \mathbf{H} matrix and the signs of the residuals are able, by themselves, to identify the set of points that are associated with masking. For real data sets, (i)–(iii) do not hold exactly. However, the masking effect is typically due to the presence of blocks of influential observations in the sample having similar or opposite effects. These blocks are likely to produce a matrix \mathbf{M} with a structure close to the one described on (i)–(iii). In fact, two influential observations i, j producing similar effects should have r_{ij} close to 1 and close to -1 when they have opposed effects. Influential observations with non-correlated effects have $|r_{ij}|$ close to 0. The same

will happen with non-influential observations. Therefore, the eigenvectors will have approximately the structure described above, and the null components will be replaced by small values. This suggests finding the eigenvectors corresponding to the p non-null eigenvalues of the influence matrix \mathbf{M} , considering the eigenvectors corresponding to large eigenvalues, and defining the sets I_j^1 and I_j^2 by those components with large positive and negative weights, respectively.

The influence matrix \mathbf{M} may be considered a generalization of Cook's local influence matrix $\mathbf{L} = \mathbf{EHE}$ (see Cook [12]). It replaces the matrix of residuals \mathbf{E} by the matrix of standardized residuals \mathbf{W} . If there are no high-leverage observations and the h_{ii} are similar for all points, both matrices will also be similar and will have similar eigenvectors. However, when the observations have very different leverage, the directions corresponding to the eigenvectors of the matrix \mathbf{M} give more weight to the influence of the high-leverage observations, which are precisely those more likely to produce masking effects.

An alternative way to analyze the information contained in the matrix of forecast changes, \mathbf{T} , is to look at its rows instead of at its columns. The rows of \mathbf{T} indicate the sensitivity of each point, that is, how the forecast of a given point is affected by all the other points in the sample. In fact the rows of this matrix

$$\mathbf{s}_i = (\widehat{y}_i - \widehat{y}_{i(1)}, \dots, \widehat{y}_i - \widehat{y}_{i(n)})'$$

describe how the forecast of the i th observation is affected by deleting any of the observations in the sample. From (31.17) we can write

$$\mathbf{s}_i = (h_{i1}e_1/(1 - h_{11}), \dots, h_{in}e_n/(1 - h_{nn})) = \mathbf{W}\mathbf{h}_i, \tag{31.22}$$

where \mathbf{W} is the diagonal matrix con elements $e_i/(1 - h_{ii})$ and \mathbf{h}_i is the i th column of \mathbf{H} . Peña and Yohai [27] define the sensitivity matrix as the covariance matrix of the sensitivity vectors of the sample points by

$$\mathbf{P} = \frac{1}{p\widehat{s}_R^2} \begin{bmatrix} \mathbf{s}'_1 \mathbf{s}_1 & \dots & \mathbf{s}'_1 \mathbf{s}_n \\ \dots & \dots & \dots \\ \mathbf{s}'_n \mathbf{s}_1 & \dots & \mathbf{s}'_n \mathbf{s}_n \end{bmatrix}.$$

with elements

$$p_{ij} = \frac{1}{p\widehat{s}_R^2} \mathbf{h}'_i \mathbf{W}^2 \mathbf{h}_j = \frac{1}{p\widehat{s}_R^2} \sum_{k=1}^n \frac{e_k^2}{(1 - h_{kk})^2} h_{ik} h_{jk},$$

which can be computed by

$$\mathbf{P} = \frac{1}{p\widehat{s}_R^2} \mathbf{H}\mathbf{W}^2\mathbf{H}. \tag{31.23}$$

It can be shown that the sensitivity and the influence matrix have both rank p and the same eigenvalues but different eigenvectors. In fact, influence and sensitivity are complementary ways to look at the observations, and the sum of the influence of all the points is equal to the sum of sensitivities. Peña and Yohai [27] have shown that the eigenvectors of the sensitivity matrix are more powerful to identify groups of outliers than those of the influence matrix and have proposed the following iterative procedure for identifying groups of outliers avoiding the masking effect:

Iteration 1: Put $k = 1$, and form a set A_1 that includes $3p + 1$ possible regression LS estimates computed as follows. The first is the standard LS estimate, computed by using all the data points. Then, for each of the p eigenvector \mathbf{z}_i , $1 \leq i \leq p$, of the \mathbf{P} matrix, with elements z_{ij} , $1 \leq j \leq n$, we obtain three other LS estimates: the first is computed dropping the points corresponding to the m largest elements z_{ij} , the second to the m smallest elements, and the third to the largest $|z_{ij}|$. Then define

$$\widehat{\boldsymbol{\beta}}^{(k)} = \arg \min_{\boldsymbol{\beta} \in A_k} S(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$$

where S is a robust measure of variability of the residuals as the MAD (median of the absolute values of the residuals) and

$$\widehat{\sigma}^{(k)} = \min_{\boldsymbol{\beta} \in A_k} S(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$$

is the value of this scale.

Iteration k : Suppose that we have selected a value $\widehat{\boldsymbol{\beta}}^{(k)}$. Compute the residuals $e_i^{(k)} = y_i - \mathbf{x}_i' \widehat{\boldsymbol{\beta}}^{(k)}$ for $1 \leq i \leq n$, and all observations with residual larger than

$$\left| e_i^{(k)} \right| > C \widehat{\sigma}^{(k)}$$

are deleted, and the rest of observations are used to a new sensitivity matrix (31.23), and its eigenvectors are used to compute the new set of estimates A_{k+1} . The procedure ends when $\widehat{\boldsymbol{\beta}}^{(k)} = \widehat{\boldsymbol{\beta}}^{(k-1)}$ and the observations deleted in the last iteration formed the set of outliers in the sample.

The values recommended by Peña and Yohai are $m = n/2$ and $C = 2$.

31.5 Robust Regression

An alternative approach to search for outliers is to use estimation methods that are robust to outliers. It is well known that LS is optimal for normal distributed data, but it may be inefficient when the data comes from a heavy tail distribution that can produce outliers. The breakdown point of an estimator is defined as the minimum fraction of outliers in the sample that makes the estimate take any arbitrary

value. For instance, the LS estimate has a breakdown point of $1/n$ because a single outlier may change $\boldsymbol{\beta}$ to any arbitrary value. If instead of LS we use a robust estimate with high breakdown point, b/n , and we have a fraction of outlier smaller than this number, b/n , we can find atypical points as observations with high residuals with respect to the robust model. These robust estimates can be obtained by, instead of finding the parameters that minimize the squared residuals as in (31.2), minimizing another function of the residuals that grows slower than the squares or, even better, that is bounded, so that the maximum effect of an outlier is limited. Robust estimation and diagnostic methods are complementary ways to analyze the problem of data heterogeneity. Diagnostic methods emphasize the need to find the possible outliers, whereas robust methods try to estimate the parameters in such a way that they are not affected by possible outliers. Both approaches complement each other, and the best procedures are often those that use ideas and results coming from both fields.

A general class of robust estimates for regression are M-estimators that minimize

$$M = \sum_{i=1}^n \rho \left(\frac{e_i(\boldsymbol{\beta})}{\widehat{\sigma}} \right) \quad (31.24)$$

where ρ is a nondecreasing bounded function with $\rho(0) = 0$, and $\rho(\infty) = 1$, so that the effect of each residual, $e_i(\boldsymbol{\beta}) = y_i - \mathbf{x}_i' \boldsymbol{\beta}$, is bounded, and $\widehat{\sigma}$ is an initial estimate of σ that makes the M-estimate independent of the scale of the observations. Particular members of this class are the LS estimate obtained, for $\rho(x) = x^2$, and the L_1 estimate, for $\rho(x) = |x|$. Note also that these two estimates do not require an initial value of $\widehat{\sigma}$. For M-estimates the value of $\widehat{\sigma}$ can be obtained by an initial estimate that does not require the scale, as, for instance, minimizing the L_1 distances

$$\sum_{i=1}^n |y_i - \mathbf{x}_i' \widehat{\boldsymbol{\beta}}^L| = \sum_{i=1}^n |e_i^L|$$

that gives an estimate more robust than LS. Then, we can obtain an initial estimate, $\widehat{\sigma}$, by the MAD (median of absolute deviations) of these residuals by

$$\widehat{\sigma} = \frac{1}{0.675} \text{Med}(|e_i^L|, e_i^L \neq 0).$$

The function (31.24) is minimized by an iterative algorithm as follows. Taking the derivative of M in (31.24) with respect to $\boldsymbol{\beta}$, the estimate must verify

$$\sum_{i=1}^n \rho' \left(\frac{e_i(\boldsymbol{\beta})}{\widehat{\sigma}} \right) \mathbf{x}_i = 0 \quad (31.25)$$

where $\rho'(x) = d\rho(x)/dx$. Now we define the weight function

$$\omega(x) = \rho'(x)/x \quad \text{if } \neq 0; \quad \omega(0) = 0$$

and the solution of (31.25) can be written as

$$\sum_{i=1}^n \omega_i(y_i - \mathbf{x}'_i \boldsymbol{\beta}) \mathbf{x}_i = 0$$

where $\omega_i = \omega(e_i(\boldsymbol{\beta})/\hat{\sigma})$; this implies

$$\boldsymbol{\beta} = \left(\sum_{i=1}^n \omega_i \mathbf{x}_i \mathbf{x}'_i \right)^{-1} \left(\sum_{i=1}^n \omega_i y_i \mathbf{x}_i \right) \quad (31.26)$$

that is a weighted LS estimate that gives a weight to each observation that depends on its standardized residual $e_i(\boldsymbol{\beta})/\hat{\sigma}$, so that observations with large residuals will have small weight in the estimation. The function (31.24) is minimized by using some initial $\boldsymbol{\beta}$ value to obtain the residuals and then computing a new value of the estimate by (31.26). This estimate will produce new residuals and new weights, and this procedure is iterated until convergence.

These M-estimates are useful when the samples only have low-leverage outliers but do not work for high-leverage outliers because, as we have seen, the residuals at these points can be very small. A better alternative is the MM-estimates in which the estimation is made in two steps: (1) an initial consistent estimate $\hat{\boldsymbol{\beta}}_0$ is computed that has high breakdown point, although it may be non-efficient, and a robust scale is obtained with this estimate; (2) Eq. (31.24) is minimized by the previous iterative procedure starting with $\hat{\boldsymbol{\beta}}_0$.

A possible initial consistent estimate is the least median of squares (LMS) estimate, proposed by Rousseeuw [42], that is computed by generating many possible values of the parameters, $\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_N$ by resampling: many random samples of size p , $(\mathbf{X}_i, \mathbf{y}_i)$ where the matrix \mathbf{X}_i is $p \times p$ and \mathbf{y}_i is $p \times 1$, are generated, and with each sample the LS estimate, $\boldsymbol{\beta}_i = \mathbf{X}_i^{-1} \mathbf{y}_i$, is computed. For each value $\boldsymbol{\beta}_i$ ($i = 1, \dots, N$), the residuals associated are obtained, $\mathbf{e}_i = \mathbf{y} - \mathbf{X} \boldsymbol{\beta}_i$ ($i = 1, \dots, N$), and the median of these residuals is used to form a robust scale

$$s(\boldsymbol{\beta}_i) = \text{median}(e_{1i}^2, \dots, e_{ni}^2). \quad (31.27)$$

The value $\boldsymbol{\beta}_i$ which minimizes this robust scale is the LMS estimate. This LMS estimate, although very robust, is not efficient. A better method is to use as initial estimate the one proposed by Peña and Yohai [27] that we have presented in Sect. 31.4.2, which can be shown has a breakdown point close to 1/2 for high-leverage concentrated outliers and p/n small. See Maronna et al. [30] for a comparison of different estimates in regression.

31.6 Detecting Outliers in Large Data Sets

The previous procedures can be very useful for moderate data sets, but they can be very slow with big data with many variables and observations. In these large data sets, iterative procedures are unfeasible and other methods are required. In this section we will recommend a simple approach and review briefly other possibilities.

A useful approach is to compute a few diagnostic statistics that can identify single and masked outliers. For masked outliers Peña [36] has proposed a single statistic that is able to identify groups of high-leverage outliers, which are the most difficult to identify. This statistic can be obtained through a proper standardization of the diagonal elements of the sensitivity matrix. Peña's P_i statistic is defined as the squared norm of the standardized sensitivity vector \mathbf{s}_i , given by (31.22), that is,

$$P_i = \frac{\mathbf{s}'_i \mathbf{s}_i}{p \widehat{\text{var}}(\hat{y}_i)}, \quad (31.28)$$

and using (31.17) and $\widehat{\text{var}}(\hat{y}_i) = \widehat{s}_R^2 h_{ii}$, this statistic can be written as

$$P_i = \frac{1}{p \widehat{s}_R^2 h_{ii}} \sum_{j=1}^n \frac{h_{ji}^2 e_j^2}{(1 - h_{jj})^2}. \quad (31.29)$$

An alternative way to write P_i is as a linear combination of the sample Cook's distance. From (31.15) and (31.29), we have

$$P_i = \sum_{j=1}^n \rho_{ji}^2 D_j, \quad (31.30)$$

where $\rho_{ij} = (h_{ij}^2/h_{ii}h_{jj})^{1/2} \leq 1$ is the correlation between forecasts \hat{y}_i and \hat{y}_j . Also, using the predictive residuals, $e_{j(j)} = e_j/(1 - h_{jj})$, we have that

$$P_i = \frac{1}{p \widehat{s}_R^2} \sum_{j=1}^n \omega_{ji} e_{j(j)}^2 \quad (31.31)$$

and P_i is a weighted combination of the predictive residuals.

The sensitivity Peña's statistics has three interesting properties. The first one is that in a sample without outliers or high-leverage observations, all the cases have the same expected sensitivity, approximately equal to $1/p$. This is an important advantage over Cook's statistic, which has an expected value that depends heavily on the leverage of the case. The second property is that for large sample sizes with many predictors, the distribution of the S_i statistic will be approximately normal. This again is an important difference from Cook's distance which has a complicated asymptotic distribution (see Muller and Mock [39]). This

normal distribution allows for computing cutoff values for finding outliers. The third property is that when the sample is contaminated by a group of similar outliers with high leverage, the sensitivity statistic will discriminate between the outliers and the good points, and the sensitivity statistic S_i is expected to be smaller for the outliers than for the good data points.

The normality of the distribution of the S_i statistic implies that we can search for outliers by finding observations with large values of $(P_i - E(P_i))/std(P_i)$. As the possible presence of outliers and high-leverage points will affect the distribution of S_i , it is better to use robust estimates as the median and MAD (median of the absolute deviations from the sample median) and consider as heterogeneous observations those which satisfy

$$|P_i - med(P)| \geq 4.5MAD(P) \quad (31.32)$$

where $med(P)$ is the median of the P_i values and $MAD(P) = median |P_i - med(P)|$. For normal data $MAD(P)/.645$ is a robust estimate for the standard deviation, and the previous rule is roughly equivalent to take three standard deviations in the normal case. See Kashif et al. [43] for other applications of this statistic.

An alternative approach is based on using the simple idea of introducing dummy variables at each point in the sample. The basic model will be

$$y_i = \mathbf{x}'_i \boldsymbol{\beta} + \sum_{j=h_1}^{h_2} w_j I_j^{(i)} + u_i, \quad i = 1, \dots, n \quad (31.33)$$

where $I_j^{(i)}$ is a dummy variable such that $I_j^{(i)} = 1$ if $j = i$ and $I_j^{(i)} = 0$ if $j \neq i$, and we allow for the possibility of an outlier at each point $h_1 = 1$ and $h_2 = n$. This model cannot be estimated because it includes $p + n$ parameters, more than the n observations, and Hendry et al. [44] and Hendry and Doornik [45] proposed the following impulse indicator saturation method: (1) Introduce dummy variables in the first half of the observations and estimate model (31.33) with $h_1 = 1$ and $h_2 = n/2$, find observations with significant w_i coefficients, and form with them a set of potential outliers. (2) Estimate model (31.33) with $h_1 = n/2 + 1$ and $h_2 = n$ in the second half of the sample, and include the outliers found in this second half in the set of potential outliers. (3) Estimate a model with dummy variables corresponding to all the observations in the set of potential outliers, and check the new estimated parameters w_i for significance. In order to control for the large number of tests made in this process, the significance level of the t test on the coefficients of the dummy variables is chosen as $\alpha = 1/n$, so that we expect to find $n\alpha = 1$ false outlier in the sample. Johansen and Nielsen [46] have shown that this procedure can be related to M regression estimates.

This method may work well when the groups of outliers occur in patches along some clear dimension, as time. If we have data ordered over time, we expect that groups of outliers will be roughly consecutive and, therefore dividing the sample into two periods of time, they will be analyzed together. Of course, we may not see the effect of a patch of outliers that occur just in the middle of the observed period, because they will be split into the two halves, but we may have an opportunity in the final estimation where dummies from both sides can be included. However, this procedure may not be very useful when there is not a clear way to split the sample and the split is made at random. Then the probability that each group of outliers appear together in one of the two halves at random will be small. We will need to repeat the procedure several times with different random splitting, but this will be unfeasible for large data sets.

A third approach is also based on model (31.33), but instead of splitting the sample, restrictions are added to estimate the parameters. These procedures still require a heavy computational burden and therefore are not suitable for very large data sets, but they are presented here because they are based on sparse analysis that may lead to powerful future procedures for big data. She and Owen [47] proposed to estimate the model assuming sparsity of the coefficients w_i and adding a Lasso penalty function

$$\min \left(\sum_{i=1}^n (y_i - \mathbf{x}'_i \boldsymbol{\beta} + w_i I_i^{(h)})^2 + \sum_{i=1}^n \lambda_i |w_i| \right),$$

and Kong, Bondell, and Wu [48] improve this approach for many variables by adding penalties on the $\boldsymbol{\beta}$ and on the w_i coefficients. Thus, they propose to estimate the model

$$\min \left(\sum_{i=1}^n (y_i - \mathbf{x}'_i \boldsymbol{\beta} + w_i I_i^{(h)})^2 + \lambda_n \sum_{i=1}^n |\beta_i| + \mu_n \sum_{i=1}^n \frac{|w_i|}{|\tilde{w}_i|} \right) \quad (31.34)$$

where \tilde{w}_i are residuals of an initial robust regression fit and λ_n and μ_n are penalty parameters. In order to implement the procedure, we need some initial robust fit, and then (31.34) is minimized by a quadratic programming algorithm.

31.7 Examples

We will show two examples in order to illustrate the previous procedures. The first is a simulated example and the second a well-known data set.

31.7.1 A Simulated Example

As an illustration we will use the simulated data from Table 31.3, which are plotted in Fig. 31.1.

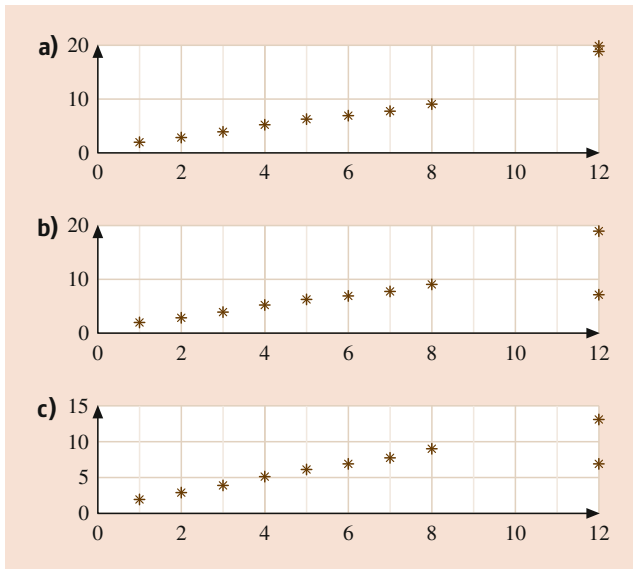


Fig. 31.1 The simulated data from Table 31.3

The three sets of data have in common cases 1 to 8 and differ in cases 9 and 10. In the first set of data, the largest values of the Cook’s statistics are $D_{10} = 0.795, D_1 = 0, 29$, and $D_9 = 0.228$. The most influential observation is the tenth which has a standardized residual $r_{10} = 1.88$; thus there is no evidence that the point is an outlier. However, the first eigenvector of the influence matrix leads to the results shown in Table 31.4. We see that both cases 9 and 10 appear separated from the rest. When they are deleted from the sample and checked against the first eight observations, we obtain the values indicated in Table 31.5, where they are clearly declared as outliers. Thus, in this example the eigenvalues of the influence matrix are able to avoid the masking effect which was clearly present in the univariate statistics.

In case (b) as both outliers have a different sign, they do not produce masking, and both of them are detected by the univariate analysis, $D_9 = 1.889$ and $D_{10} = 1.893$, and the outlier tests are $t_{10} = 5.20$ and $t_9 = -5.24$. The two points are also shown in the extremes of the eigenvalue. Finally in case (c) there is only an outlier which is detected by both the univariate and multivariate analysis.

Table 31.3 A simulated set of data from which three different samples are generated

	1	2	3	4	5	6	7	8	9(a)	10(a)	9(b)	10(b)	9(c)	10(c)
x	1	2	3	4	5	6	7	8	12	12	12	12	12	12
y	2.0	2.9	3.9	5.1	6.2	6.9	7.8	9.1	19	20	19	7	13	7

Table 31.4 Eigenanalysis of the influence matrix for the three samples from Table 31.3. The first two eigenvalues and the first eigenvector are shown

	λ_1	λ_1/λ_2	1	2	3	4	5	6	7	8	9	10
(a)	1.27	2.87	-.17	-.06	-.00	-.00	-.02	-.10	-.22	-.33	.42	.79
(b)	3.78	3.783	.00	-.00	-.00	-.00	-.00	.00	-.00	-.00	-.71	.71
(c)	3.25	32	-.05	-.02	-.00	-.00	-.01	-.02	-.04	-.10	-.50	.85

Table 31.5 Values of the t statistic for testing each point as an outlier

Case	9	10
(a)	27.69	32.28
(b)	31.94	-32.09
(c)	-0.07	-32.09

If we compute the eigenvectors of the sensitivity matrix for these data, we obtain the results presented in Table 31.6. The first eigenvector clearly separates the observations 9 and 10 from the rest. In fact, if we order the coordinates of this vector, we find the largest ratio at $170/22 = 8.5$ which separates cases 9 and 10 from the others.

31.7.2 The Housing Boston Data

We analyze the Boston housing data set which consists of 506 observations on 14 variables, available at <http://lib.stat.cmu.edu/datasets/boston>. This data set was given by Belsley et al. [3] and we have used the same variables they considered: The dependent variable is the logarithm of the median value of owner-occupied homes.

Figure 31.2 shows the diagnostic analysis of this data set. The first row corresponds to the residuals of the regression model. The residuals have been divided by their standard error, and the first plot shows a few points which can be considered as outliers. The plot of the studentized residual is similar and identifies the same points as outliers. The second row gives information about the Cook’s D statistics. There are clearly some points in the middle of the sample which are more influential than the rest, but all the values of the statistic are small and, as we expect a skew distribution, the conclusion is not clear. However, the sensitivity statistics clearly identifies a group of extreme observations which are not homogeneous with the rest. The median of the sensitivity statistic is 0.0762, very close to the expected value $1/p = 1/14 = 0.0714$. The MEDA is 0.0195 and the plot indicates that 45 observations are heterogeneous with respect to the rest. These observations are most of the cases 366–425 and some other isolated points. From Belsley et al. [3], we obtain that cases 357–488 correspond to Boston, whereas the rest correspond to the suburbs. Also, the 45 points indicated by

Table 31.6 Eigenvalues of the sensitivity matrix for data from Table 31.3

	1	2	3	4	5	6	7	8	9	10
v_1	0.502	0.455	0.407	0.360	0.312	0.264	0.217	0.170	-0.020	-0.020
v_2	-0.191	-0.119	-0.046	0.026	0.099	0.172	0.245	0.318	0.610	0.610

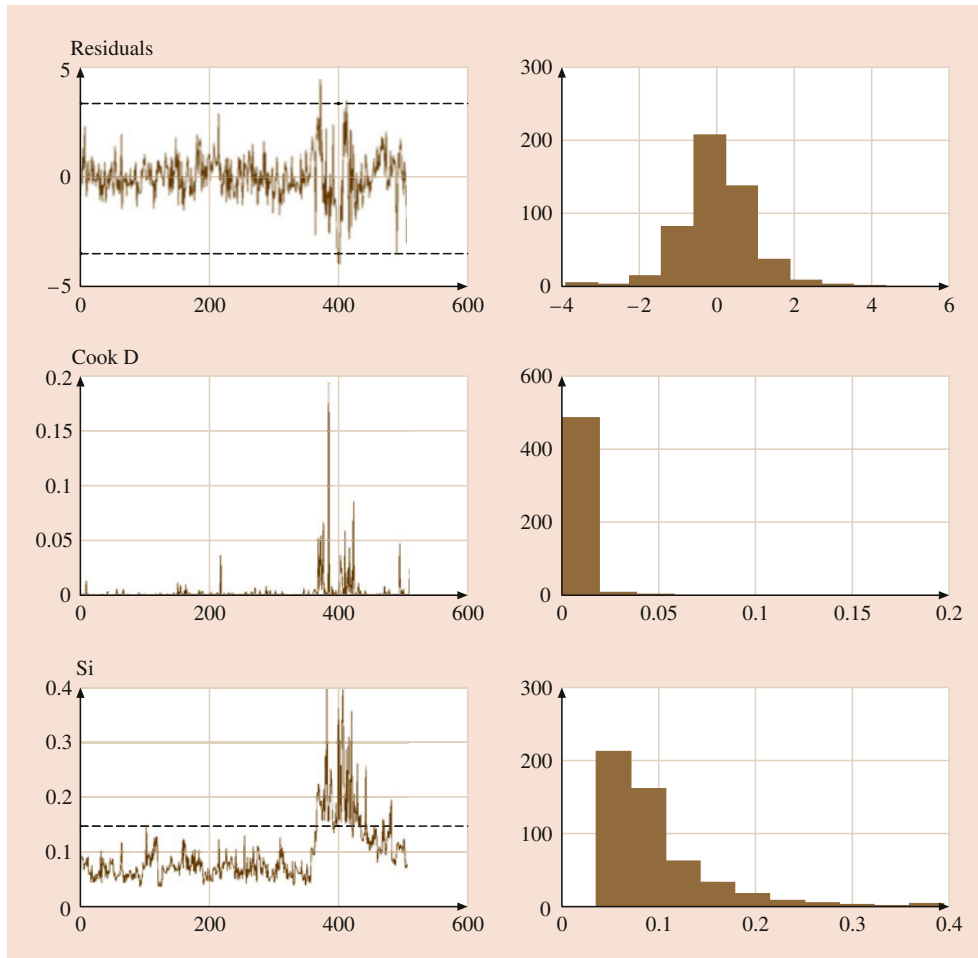


Fig. 31.2 Residuals, Cook’s statistics, and sensitivity statistics for the Boston housing data. Right, histogram; left, case plot of the value of the statistic

statistic S_i as outliers all correspond to some central districts of Boston, including downtown, which suggests that the relation among the variables could be different in these districts than in the rest of the sample. In fact, if we fit regression equations to these two groups, we find very different coefficients for the regression coefficients in both groups of data, and in the second group, only five variables are significant. Also, we obtain a large reduction in residual sum of squares (RSE) when fitting different regression equations in the two groups.

Figure 31.3 shows the first eigenvalues of the matrix of influence and sensitivity. Although both eigenvectors indicate the heterogeneity, the one from the matrix of sensitivity is more clear.

The procedure of impulse indicator saturation identifies 54 outliers in the second half of the observations correspond-

ing to the downtown area. In this case the procedure works well because the group of consecutive outliers is in agreement with the sequence of the observations.

31.8 Final Remarks

We have shown different procedures for diagnosis in regression model and have stressed that the detection of groups of outliers in regression in large data sets can be made by the eigenanalysis of the influence and sensitivity matrices. We have also shown that a single statistic of sensitivity is able to reveal masked outliers in large data set. The most challenging problem today is to identify heterogeneity when we do not have a central model which explains more than 50% of the data and some groups of outliers, as it has been

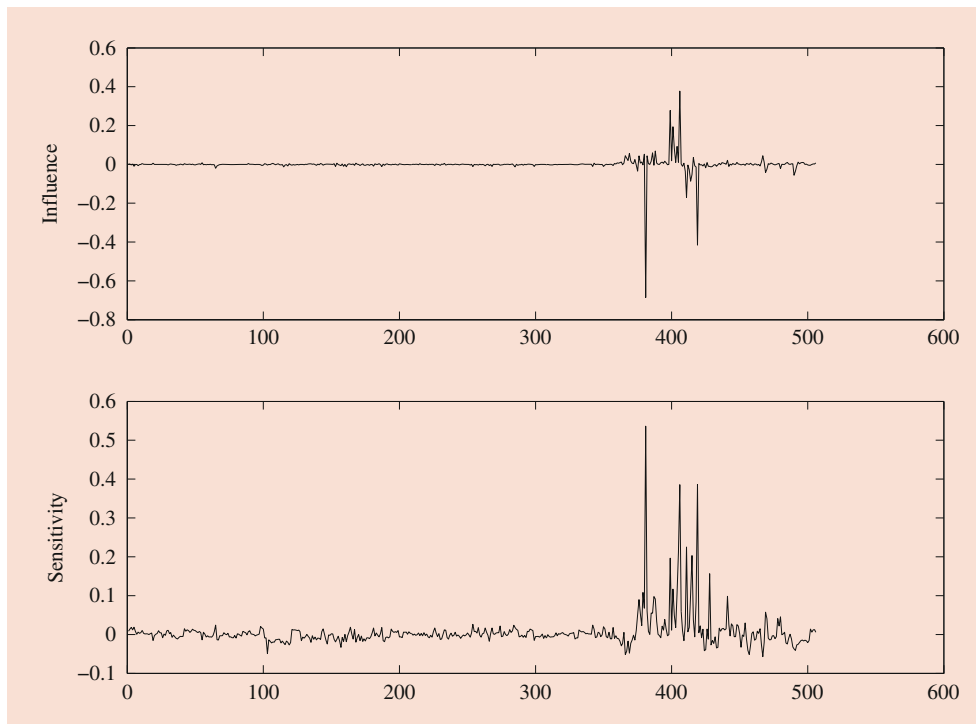


Fig. 31.3 First eigenvalue of the influence and sensitivity matrices

assumed in this chapter, but different regression models in different regions of the parameter space. In this case the robust methods are no longer useful and we need other methods to solve this problem. These situations are very close to cluster analysis, and [10] showed the usefulness of clustering procedures in detecting outliers. In fact, if we apply *k*-means to the housing data analyzed in Sect. 31.7, the two groups of regression are found. Better cluster procedures can be developed by searching directly for clusters around different regression lines; see Peña, Rodríguez, and Tiao [49] and García-Escudero et al. [50, 51] for different approaches to this problem.

Acknowledgments This research has been supported by Grant ECO2015-66593-P of MINECO/FEDER/UE, Spain.

References

1. Box, G.E.P.: When Murphy speaks listen. *Qual. Prog.* **22**, 79–84 (1989)
2. Hampel, F.R., Ronchetti, E.M., Rousseeuw, P.J., Stahel, W.A.: *Robust Statistics*. Wiley, New York (1986)
3. Belsley, D.A., Kuh, E., Welsch, R.E.: *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. Wiley, New York (1980)
4. Hawkins, D.M.: *Identification of Outliers*. Chapman Hall, New York (1980)
5. Cook, R.D., Weisberg, S.: *Residuals and Influence in Regression*. Chapman Hall, New York (1982)
6. Atkinson, A.C.: *Plots, Transformations and Regression*. Clarendon, Oxford (1985)
7. Chatterjee, S., Hadi, A.S.: *Sensitivity Analysis in Linear Regression*. Wiley, New York (1988)
8. Barnett, V., Lewis, T.: *Outliers in Statistical Data*, 3 edn. Wiley, New York (1994)
9. Atkinson, A.C., Riani, M.: *Robust Diagnostic Regression Analysis*. Springer, Berlin, Heidelberg, New York (2012)
10. Carroll, R.J.: *Transformation and Weighting in Regression*. Routledge (2017)
11. Cook, R.D.: Detection of influential observations in linear regression. *Technometrics* **19**, 15–18 (1977)
12. Cook, R.D.: Assessment of local influence (with discussion). *J. R. Stat. Soc. B* **48**(2), 133–169 (1986)
13. Suárez Rancel, M., González Sierra, M.A.: Regression diagnostic using local influence: A review. *Commun. Stat. A* **30**, 799–813 (2001)
14. Hartless, G., Booth, J.G., Littell, R.C.: Local influence of predictors in multiple linear regression. *Technometrics* **45**, 326–332 (2003)
15. Critchley, F., Atkinson, R.A., Lu, G., Biazzi, E.: Influence analysis based on the case sensitivity function. *J. R. Stat. Soc. B* **63**(2), 307–323 (2001)
16. Lawrance, J.: Deletion influence and masking in regression. *J. R. Stat. Soc. B* **57**, 181–189 (1995)
17. Hawkins, D.M., Bradu, D., Kass, G.V.: Location of several outliers in multiple regression data using elemental sets. *Technometrics* **26**, 197–208 (1984)
18. Gray, J.B., Ling, R.F.: *K*-Clustering as a detection tool for influential subsets in regression. *Technometrics* **26**, 305–330 (1984)
19. Marasinghe, M.G.: A multistage procedure for detecting several outliers in linear regression. *Technometrics* **27**, 395–399 (1985)
20. Kianifard, F., Swallow, W.: Using recursive residuals calculated in adaptively ordered observations to identify outliers in linear regression. *Biometrics* **45**, 571–585 (1989)

21. Kianifard, F., Swallow, W.: A Monte Carlo comparison of five procedures for identifying outliers in lineal regression. *Commun. Stat. (Theory and Methods)* **19**, 1913–1938 (1990)
22. Hadi, A.S., Simonoff, J.S.: Procedures for the identification of multiple outliers in linear models. *J. Am. Stat. Assoc.* **88**, 1264–1272 (1993)
23. Hadi, A.S., Simonoff, J.S.: Improving the estimation and outlier identification properties of the least median of squares and minimum volume ellipsoid estimators. *Parisankhyan Samikkha* **1**, 61–70 (1994)
24. Atkinson, A.C.: Fast very robust methods for the detection of multiple outliers. *J. Am. Stat. Assoc.* **89**, 1329–1339 (1994)
25. Swallow, W., Kianifard, F.: Using robust scale estimates in detecting multiple outliers in linear regression. *Biometrics* **52**, 545–556 (1996)
26. Peña, D., Yohai, V.J.: The detection of influential subsets in linear regression using an influence matrix. *J. R. Stat. Soc. B* **57**, 145–156 (1995)
27. Peña, D., Yohai, V.J.: A fast procedure for robust estimation and diagnostics in large regression problems. *J. Am. Stat. Assoc.* **94**, 434–445 (1999)
28. Huber, P.: Between Robustness and Diagnosis. In: Stahel, W., Weisberg, S. *Directions in Robust Statistics and Diagnosis*, pp. 121–130. Springer, Berlin, Heidelberg, New York (1991)
29. Rousseeuw, P.J., Leroy, A.M.: *Robust Regression and Outlier Detection*. Wiley, New York (1987)
30. Maronna, R.A., Martin, R.D., Yohai, V.J., Saliban-Barrera, M.: *Robust Statistics, Theory and Methods (with R)*. Wiley, New York (2019)
31. Box, G.E.P., Tiao, C.G.: A Bayesian approach to some outlier problems. *Biometrika* **55**, 119–129 (1968)
32. Peña, D., Guttman, I.: Comparing probabilistic models for outlier detection. *Biometrika* **80**(3), 603–610 (1993)
33. Berger, J.O., Moreno, E., Pericchi, L.R., Bayarri, M.J., Bernardo, J.M., Cano, J.A., ..., Dasgupta, A.: An overview of robust Bayesian analysis. *Test* **3**(1), 5–124 (1994)
34. Justel, A., Peña, D.: Bayesian unmasking in linear models. *Comput. Stat. Data Anal.* **36**, 69–94 (2001)
35. Hans, C.: Bayesian lasso regression. *Biometrika* **96**(4), 835–845 (2009)
36. Peña, D.: A new statistic for influence in linear regression. *Technometrics* **47**(1), 1–12 (2005)
37. Diaz-García, J.A., Gonzalez-Farias, G.: A note on the Cook distance. *J. Stat. Plann. Infer.* **120**, 119–136 (2004)
38. Cook, R.D., Peña, D., Weisberg, S.: The likelihood displacement. A unifying principle for influence. *Commun. Stat. A* **17**, 623–640 (1988)
39. Muller, E.K., Mok, M.C.: The distribution of Cook D statistics. *Commun. Stat. A* **26**, 525–546 (1997)
40. Atkinson, A.C.: Masking unmasked. *Biometrika* **73**, 533–41 (1986)
41. Wisnowski, J.W., Montgomey, D.C., Simpson, J.R.: A comparative analysis of multiple outliers detection procedures in the linear regression model. *Comput. Stat. Data Anal.* **36**, 351–382 (2001)
42. Rousseeuw, P.J.: Least median of squares regression. *J. Am. Stat. Assoc.* **79**, 871–880 (1984)
43. Kashif, M., Amanullah, M., Aslam, M.: Pena’s statistic for the Liu regression. *J. Stat. Comput. Simul.* **88**(13), 2473–2488 (2018)
44. Hendry, D.F., Johansen, S., Santos, C.: Automatic selection of indicators in a fully saturated regression. *Computational Statistics* **33**, 317–335 (2008); Erratum, 337–339
45. Hendry, D.F., Doornik, J.A.: *Empirical Model Discovery and Theory Evaluation: Automatic Selection Methods in Econometrics*. MIT Press (2014)
46. Johansen, S., Nielsen, B.: Asymptotic theory of outlier detection algorithms for linear time series regression models. *Scand. J. Stat.* **43**(2), 321–348 (2016)
47. She, Y., Owen, A.B.: Outlier detection using nonconvex penalized regression. *J. Am. Stat. Assoc.* **106**(494), 626–639 (2011)
48. Kong, D., Bondell, H., Wu, Y.: Fully efficient robust estimation, outlier detection and variable selection via penalized regression. *Statistica Sinica* **28**, 1031–1052, (2018).
49. Peña, D., Rodriguez, J., Tiao, G.C.: Identifying mixtures of regression equations by the SAR procedure (with discussion). In: Bernardo et al., *Bayesian Statistics*, vol. 7, pp. 327–347. Oxford Univ. Press, Oxford (2003)
50. García-Escudero, L.A., Gordaliza, A., San Martín, R., Van Aelst, S., Zamar, R.: Robust linear clustering. *J. Roy. Stat. Soc. Ser. B (Statistical Methodology)* **71**(1), 301–318 (2009)
51. García-Escudero, L.A., Gordaliza, A., Mayo-Isacar, A., San Martín, R.: Robust clusterwise linear regression through trimming. *Comput. Stat. Data Anal.* **54**(12), 3057–3069 (2010)



Daniel Peña is Professor, Department of Statistics and Big Data Institute, Universidad Carlos III de Madrid. He is the author of 13 books and more than 200 research papers. Fellow of the American Statistical Association, The Institute of Mathematical Statistics, Elected member of the International Statistical Institute and recipient of the Youden Prize and the Jaime I Award for his research contributions.



Statistical Methodologies for Analyzing Genomic Data

32

Fenghai Duan and Heping Zhang

Contents

32.1	Introduction	622
32.2	Second-Level Analysis of Microarray Data	623
32.2.1	Notation.....	623
32.2.2	Fold Change.....	623
32.2.3	<i>t</i> -Statistic.....	623
32.2.4	The Multiple Comparison Issue.....	623
32.2.5	Empirical Bayesian Approach.....	624
32.2.6	Significance Analysis of Microarray (SAM).....	624
32.3	Third-Level Analysis of Microarray Data	625
32.3.1	Clustering.....	625
32.3.2	Classification.....	628
32.3.3	Tree- and Forest-Based Classification.....	630
32.4	Fourth-Level Analysis of Microarray Data	632
32.5	Final Remarks	632
	References	632

Abstract

The purpose of this chapter is to describe and review a variety of statistical issues and methods related to the analysis of microarray data. In the first section, after a brief introduction of the DNA microarray technology in biochemical and genetic research, we provide an overview of four levels of statistical analyses. The subsequent sections present the methods and algorithms in detail.

In the second section, we describe the methods for identifying significantly differentially expressed genes in

different groups. The methods include fold change, different *t*-statistics, empirical Bayesian approach, and significance analysis of microarrays (SAM). We further illustrate SAM using a publicly available colon cancer dataset as an example. We also discuss multiple comparison issues and the use of false discovery rate.

In the third section, we present various algorithms and approaches for studying the relationship among genes, particularly clustering and classification. In clustering analysis, we discuss hierarchical clustering, and *k*-means and probabilistic model-based clustering in detail with examples. We also describe the adjusted Rand index as a measure of agreement between different clustering methods. In classification analysis, we first define some basic concepts related to classification. Then we describe four commonly used classification methods including linear discriminant analysis (LDA), support vector machines (SVM), neural network, and tree-and-forest-based classification. Examples are included to illustrate SVM and tree-and-forest-based classification.

The fourth section is a brief description of the meta-analysis of microarray data in three different settings: meta-analysis of the same biomolecule and same platform microarray data, meta-analysis of the same biomolecule but different platform microarray data, and meta-analysis of different biomolecule microarray data.

We end this chapter with final remarks on future prospects of microarray data analysis.

F. Duan
Department of Biostatistics, Brown University School of Public Health, Providence, RI, USA
e-mail: fduan@stat.brown.edu

H. Zhang (✉)
Department of Biostatistics, Yale University School of Public Health, New Haven, CT, USA
e-mail: heping.zhang@yale.edu

Keywords

Support vector machine · Artificial neural network · False discovery rate · Microarray data · Random forest

32.1 Introduction

Since the seminal work on microarray technology of *Schena* et al. [1], microarray data have attracted a great deal of attention, as reflected by the ever-increasing number of publications on this technology in the past decade. The applications of the microarray technology encompass many fields of science from the search for differentially expressed genes [2] to the understanding of regulatory networks [3], DNA sequencing and mutation study [4], single-nucleotide polymorphism (SNP) detection [5], cancer diagnosis [6], and drug discovery [7].

Accompanying the advancement of the microarray technology, analyzing microarray data has arguably become the most active research area of statistics and bioinformatics. Figure 32.1 provides a four-level overview of the analytic process. The first challenge in dealing with the microarray data is to preprocess the data, which involves background subtraction, array normalization, and probe-level data summarization. The purpose of this preprocessing is to remove noise and artifacts in order to enhance and extract hybridization signals. This data preprocessing is also often referred as the low-level analysis [8]. After the data are processed

and cleaned, they are analyzed for different purposes. The focus of this article is on the methods for this postprocessing analysis.

The second-level analysis usually contains two steps: one is to filter *unusual* genes whose expression profiles are suspicious due to noise or are too extreme, and the other is to identify the differentially expressed genes across different samples. The gene filtration process is generally heuristic and specific to known biological contents. Thus, we will not discuss it here. To identify genes that have significantly different expression profiles, the commonly used approaches include the estimation of fold change, Student's *t*-test, the Wilcoxon rank sum test, the penalized *t*-test, empirical Bayes [9], and significance analysis of microarray (SAM, *Tusher* et al. [10]). We will review these methods in Sect. 32.2.

We will review the third-level analysis in Sect. 32.3. This type of analysis is also called high-level analysis [11], and it includes clustering, classification, and pathway analysis. This is usually conducted on a subset of genes that are selected from the second-level analysis. To identify genes that may be correlated to each other, clustering analysis has become particularly popular, and the approaches include hierarchical clustering [12], *k*-means [13], self-organization maps (SOM) [14], principle component analysis (PCA) [15], and probabilistic model-based clustering [16].

To classify tissue samples or diagnose diseases based on gene expression profiles, both classic discriminant analysis and contemporary classification methods have been used and developed. The methods include *k*-nearest neighbors (KNN) [17], linear discriminant analysis (LDA) [18], support vector machine (SVM) [19], artificial neural networks (ANN) [20], classification trees [21], and random and deterministic forests [18]. It is noteworthy that tree- and forest-based approaches can be easily applied to the entire microarray dataset without restricting our attention to a subset of selected genes.

To identify genes that may be on the same pathway of a particular biological process, relevance networks [22], linear differential equation [23], Boolean networks [24], Bayesian networks [25], and the probabilistic rational model (PRM) [26] have been used and developed.

The fourth-level analysis, also referred as meta-analysis, is a relatively new topic for the analysis of microarray data. Because many different types and platforms of microarrays can be designed to address the same (or similar) biological problems, it is useful to compare and synthesize the results from different studies.

Before we introduce specific methods, we should point out that, as a result of high-throughput technology, the unique challenge from analyzing microarray data is the large number of genes (tens of thousands) and relatively small sample sizes (commonly on the order of tens or hundreds). In this article, n denotes the number of genes and m the number of arrays. n is generally much greater than m .

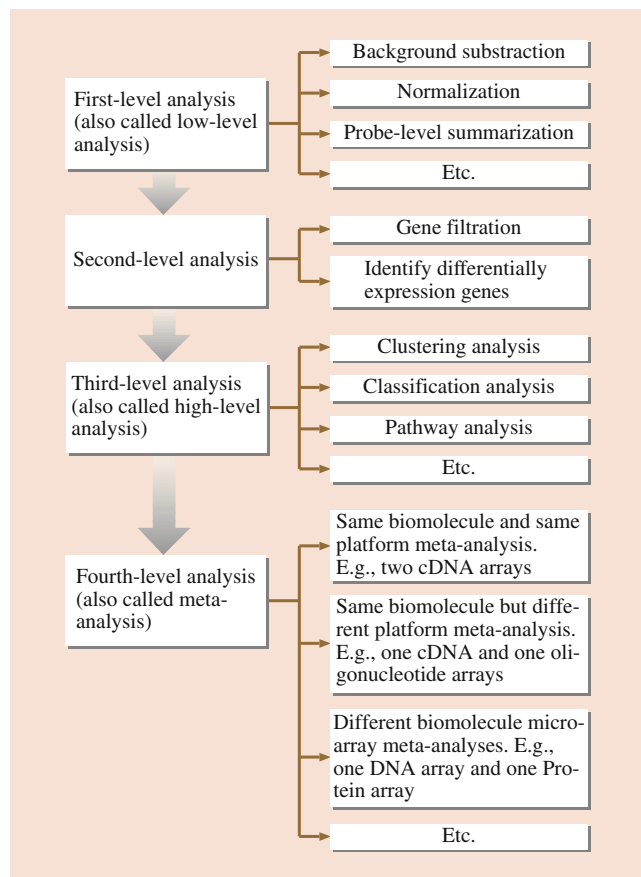


Fig. 32.1 Diagram of the four-level analysis of microarray data

32.2 Second-Level Analysis of Microarray Data

32.2.1 Notation

For a two-channel cDNA microarray data [1], we have a $2n \times m$ matrix of imaging data reflecting the red (cy5) and green (cy3) signals for each of the n genes on m arrays. The log ratio of the red to green signal is usually taken for each gene, and the analysis will be based on an $n \times m$ data matrix.

For one-channel Affymetrix Oligonucleotide Gene-Chip data [27], we have a $2 \sum_{i=1}^n p_i \times m$ matrix of raw image data where p_i is the number of probes for the i -th gene. Note that, for each probeset, Affymetrix uses a pair of perfect match (PM) and mismatch (MM). As for oligonucleotide microarrays, steps (differences, ratios, analysis of variance (ANOVA) models, etc.) can be taken to summarize the PM and MM signals for each gene, and we still have an $n \times m$ data matrix.

A major objective of microarray analysis is to infer significantly differentially expressed genes (abbreviated as SDE genes) across different samples, e.g., m_1 tumor samples versus m_2 normal samples.

Let $Y_{ij,k}$ be the expression level of the i -th gene on the j -th array in the k -th sample. Let $\bar{Y}_{i.,1}$ and $\bar{Y}_{i.,2}$ denote the average expression level of the i -th gene in samples 1 and 2, respectively.

32.2.2 Fold Change

Many studies identify SDE genes in two samples based on simple fold-change thresholds such as a two-fold change in means. Although the choice of a threshold is somewhat arbitrary, fold change is intuitive and biologically meaningful, and serves as an effective preliminary step to eliminate a large portion of genes whose data are of little interest in a particular study.

32.2.3 t -Statistic

As in many clinical studies, the t -statistic provides a simple, extremely useful tool to compare the data from two samples. Let \bar{M} be the mean difference between the expression profiles of a gene in two groups and $\text{se}(\bar{M})$ be the standard error of \bar{M} . The t -statistic, defined as

$$t = \frac{\bar{M}}{\text{sd}(\bar{M})},$$

is useful to test a null hypothesis that the gene is not differentially expressed in the two groups against the alternative hypothesis that the gene is differentially expressed.

Unlike a typical clinical study, in which we have one pair or a very few pairs of hypotheses to test, in microarray analysis we have a pair of hypotheses for every gene of interest. This means that we inevitably deal with the multiple comparison issue. Although this issue is difficult and there is no clear-cut, ideal answer, many reasonable solutions have been proposed.

Efron et al. [9] proposed to inflate $\text{se}(\bar{M})$ by adding a constant that equals the 90-th percentile of the standard errors of all the genes. *Tusher* et al. [10] call such a constant a fudge factor, and propose to estimate it by minimizing the coefficient of variation of the absolute t -values. We will discuss this approach in detail in Sect. 32.2.4. Other approaches have also been proposed; for example, *Smyth* [28] replaces $\text{se}(\bar{M})$ with a Bayesian shrinkage estimator of the standard deviation.

The permutation test is also commonly used to compare the microarrays. Permutations are usually performed at the array level to create a situation similar to the null hypothesis while maintaining the dependence structure among the genes [10]. In every permutation, a t -statistic can be calculated for each gene. Once a large number of permutations are completed, we have an empirical distribution for the t -statistic under the null hypothesis, which then can be used to identify SDE genes.

32.2.4 The Multiple Comparison Issue

As we mentioned earlier, we have to control the type I error rate α while testing a large number of hypotheses simultaneously. There are two commonly used approaches to deal with this issue. One is to control the family-wise error rate (FWER) and the other is to control the false discovery rate (FDR).

The FWER controls the probability of making at least one false positive call at the desired significance level. FWER guarantees that the type I error rate is less than or equal to a specified value for any given set of genes. The most known example of FWER is Bonferroni correction that divides the desired significance level α by total number of hypotheses. If the desired significance level is 0.05 and we compare expression profiles in 10,000 genes, a gene is declared to have significantly different profiles in two groups if the P -value is not greater than $\frac{0.05}{10,000} = 5 \times 10^{-6}$. Another FWER approach is the so-called *Šidák* correction in which the adjusted type I error rate is at $1 - (1 - \alpha)^{\frac{1}{n}}$ [29], which is close to α/n . Clearly, Bonferroni and *Šidák* corrections are sufficient but not necessary conditions [30], and FWER approaches are generally very conservative and set a stringent bar to declare SDE genes.

Because of the conservative nature of the FWER approaches, the FDR concept has flourished since it was proposed by [31]. FDR is defined as the mean of the ratio of the

number, denoted by V , of falsely rejected hypotheses to the total rejected hypotheses, denoted by R , namely,

$$\text{FDR} = E \left(\frac{V}{R} \mid R > 0 \right) \Pr(R > 0),$$

where $\Pr(R > 0)$ is the probability of rejecting at least one hypothesis.

The FDR can be controlled at a given α level through the following steps. First, for n genes, we have n null hypotheses and np values, denoted by p_1, \dots, p_n . Then, we sort the p -values in ascending order such that $p_{(1)} \leq \dots \leq p_{(n)}$. We reject any gene i that satisfies the condition $p(i) \leq \frac{i}{n} \times \frac{\alpha}{p_0}$, where p_0 is the proportion of genes for which the null hypotheses are indeed true. Because p_0 is unknown in practice, the most conservative approach is to replace it with 1. Recently, attempts have been made to estimate p_0 as in Tusher et al.'s SAM, where they used a permutation procedure to estimate p_0 . Similar to the classical p -values, the significance measures for each gene in terms of FDR are called q -values, a name that was introduced by Storey [32, 33].

In addition, the FDR concept has been generalized. For example, Storey and Tibshirani [9] and Storey et al. [32] proposed positive FDR (pFDR), which corrects the error rate only when there are positive findings. For microarray data, many gene profiles are correlated; Troendle [34] proposed an adjusted FDR to address the correlation and demonstrated the benefit in terms of gained power.

32.2.5 Empirical Bayesian Approach

Using microarray data from a breast cancer study, Efron et al. [9, 35] described the empirical Bayesian method. As an initial step, a summary statistic, Z , needs to be defined for every gene to reflect the scientific interest; this can be the t -statistic as described above, a Wilcoxon rank statistic, or another choice. All genes are perceived to belong to either the differentially or nondifferentially expressed group. The density of Z_i is $f_0(z_i)$ if gene i is in the nondifferentially expressed group, and $f_1(z_i)$ otherwise. Without knowing the group, Z_i has the following mixture distribution:

$$p_0 f_0(z_i) + p_1 f_1(z_i),$$

where p_0 is the prior probability that gene i is not differentially expressed, and $p_1 = 1 - p_0$.

Based on Bayes' theorem, the posterior probability that gene i is not differentially expressed given Z_i is

$$p_0(z_i) = p_0 \frac{f_0(z_i)}{f(z_i)}.$$

We can estimate the mixture density $f(z_i)$ by the empirical distribution $\hat{f}(z_i)$ because the genes of interest are naturally a mixture of the two groups. In addition, the null density $f_0(z_i)$ can be estimated through the permutation that artificially generates data under the null hypothesis. In other words, we can derive the posterior probability $p_0(z_i)$.

For a given prior p_0 . The choice of p_0 can be subjective. One conservative possibility is to choose p_0 to be the minimum of $\hat{f}(z_i) / \hat{f}_0(z_i)$ so that the posterior probability $p_1(z_i)$ that gene i is differentially expressed is non-negative. Note that $p_1(z_i) = 1 - p_0(z_i)$. Finally, all genes can be ranked according to $p_1(z_i)$ and highly probably differentially expressed genes can be selected.

Efron et al. [9, 35] did not assume a specific form for $f(z_i)$. In contrast, Lonnstedt and Speed [36] assumed that the data comes from the mixture of normal distributions and used the conjugate priors for the variances and the means. Under those assumptions, they derived the log odds posterior test. Smyth [28] extended the hierarchical model of Lonnstedt and Speed [36] to deal with microarray experiments with more than two sample groups. The method is called the Limma algorithm.

32.2.6 Significance Analysis of Microarray (SAM)

Tusher et al. [10] introduced the SAM algorithm. SAM identifies genes with statistically significant changes in expression by assimilating a set of gene-specific t -tests in which the standard error is adjusted by adding a small positive constant. It performs a random permutation among experiments and declares the significant genes based on a selected threshold. For the given threshold, SAM estimates the FDR by comparing the number of genes significant in the permuted samples with the number of genes significant in the original sample.

SAM can be downloaded from <http://www-stat.stanford.edu/~tibs/SAM/>. Specifically, first, for each gene i , SAM computes a t -like statistic

$$t_i = \frac{r_i}{s_i + s_0},$$

where r_i is the difference between the expression means of gene i in the two groups (expression is on a logarithm scale), s_i is the standard error, and s_0 is the fudge factor to be estimated. Secondly, similarly to the FDR scheme, all t_i values are sorted into the order statistics

$$t_{(1)} \leq t_{(2)} \leq \dots \leq t_{(n)}.$$

To choose the significance threshold, the expression data are permuted in the two groups within each gene B times, and during each permutation, we repeat the first two steps, which leads to a set of order statistics:

$$t_{(1)}^b \leq t_{(2)}^b \leq \dots \leq t_{(n)}^b.$$

After the permutations, we calculate the mean of the order statistics for each gene as follows:

$$\bar{t}_{(i)} = \frac{1}{B} \sum_{b=1}^B t_{(i)}^b.$$

For a given threshold Δ , a gene is considered significant if $|t_{(i)} - \bar{t}_{(i)}| > \Delta$, and the FDR is estimated by the ratio of the number of genes found to be significant in the permutation samples to the number of genes called significant in the original sample.

Example 1: Identification of SDE Genes Using SAM

In this example, we apply SAM to examine a publicly available colon cancer dataset [37]. This dataset contains the expression profiles of 2000 genes using an Affymetrix oligonucleotide array in 22 normal and 40 colon cancer tissues.

Figure 32.2 displays the quantile–quantile plot from SAM. The two dashed lines determine a boundary to call genes SDE depending on the choice of Δ . For example, Δ was chosen as 0.9857 in Fig. 32.2 to control the FDR at about 5%. The white square and triangle points in the figure correspond to the genes that are declared to be significantly overexpressed and underexpressed, respectively. Out of the 490 declared SDE genes (440 overexpressed and 50 underexpressed), 25 genes are expected to be declared falsely.

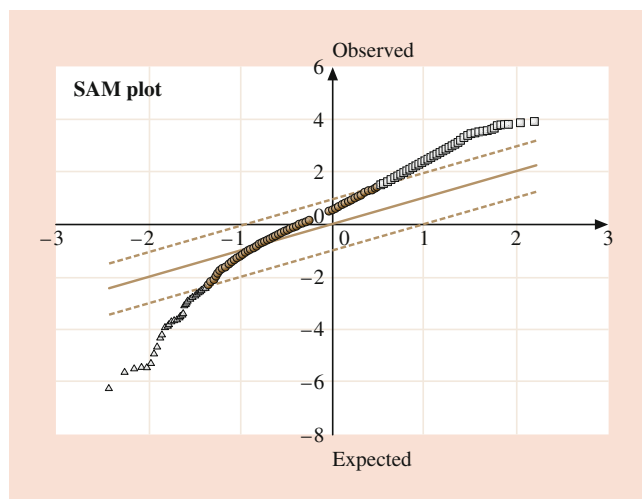


Fig. 32.2 The quantile–quantile plot from SAM for the colon cancer dataset. Genes are declared significantly changed when their corresponding t -values are outside the two dashed lines. The white square and triangle points correspond to the genes that are significantly overexpressed and underexpressed, respectively

32.3 Third-Level Analysis of Microarray Data

The third-level microarray analysis includes clustering, classification, and pathway analysis. These approaches usually, though not always, follow the second-level microarray analysis because most of them can work effectively on only a small number of genes.

32.3.1 Clustering

Clustering is arguably the most commonly used approach at the third level of analysis [38, 39]. It is an unsupervised learning algorithm from a machine learning viewpoint, because the gene classes are unknown or not used, and need to be *discovered* from the data. Therefore, the goal of clustering analysis is to group genes (or arrays) based on their similarity in the feature space (e.g., expression pattern).

The underlying assumption behind clustering is that genes with similar expression profiles should share some common biological behaviors, e.g., belonging to the same protein complex or gene family [40], having common biological functions [41], being regulated by common transcription factors [3], belonging to the same genetic pathway, or coming from the same origin [39].

After the clusters are formed, a dendrogram or a tree of all genes will be viewed, although the views are not unique, because there is a left-or-right selection at each splitting step. Two popular programs for gene clustering are Eisen et al.’s TreeView program [12] and Li and Wong’s dChip programs [8]. Routines are also available in standard statistical packages such as R, Splus, and SAS.

Distance

In order to group objects (genes or arrays) together, we need to define a measure to quantify the similarity among objects in the feature space. Such a measure of similarity is called a distance. There are several commonly used definitions of distance. Suppose that the expression profiles of two genes are $Y_i = (y_{i1}, y_{i2}, \dots, y_{im})$ and $Y_j = (y_{j1}, y_{j2}, \dots, y_{jm})$.

The Euclidean distance between Y_i and Y_j is

$$d_E(Y_i, Y_j) = \left[\sum_{k=1}^m (y_{ik} - y_{jk})^2 \right]^{\frac{1}{2}}.$$

The city-block distance between Y_i and Y_j is

$$d_C(Y_i, Y_j) = \sum_{k=1}^m |y_{i1} - y_{j1}|.$$

The Pearson correlation distance between Y_i and Y_j is

$$d_R(Y_i, Y_j) = 1 - r_{Y_i Y_j},$$

where $r_{Y_i Y_j}$ is the Pearson correlation coefficient between Y_i and Y_j .

The Spearman correlation distance between Y_i and Y_j uses the rank-based correlation coefficient in which the expression levels are replaced with the ranks.

More definitions can be found in the book by *Draghici* [30]. We should note that the Euclidean and city-block distance look for similar expression numerical values while the Pearson and Spearman distances tend to emphasize similar expression patterns.

The distances defined above measure the gene-wise distance. When clusters are found, we also need to define the distance between two clusters. The four approaches are: single linkage distance (the minimum distance between any gene in one cluster and any gene in the other cluster), complete linkage distance (the maximum distance between any gene in one cluster and any gene in the other cluster), average linkage distance (the average of all pair-wise distances between any gene in one cluster and any gene in the other cluster), and centroid linkage distance (the distance between the centroids of the two clusters).

Clustering Methods

When a distance measure is chosen, there are different ways to execute the clustering process. The clustering methods broadly fall into two categories: hierarchical methods and partitioning methods. Hierarchical methods build up a hierarchy for clusters, from the lowest one (all genes are in one cluster) to the highest one (all genes are in their own clusters), while partitioning methods group the genes into the different clusters based on their expression profiles. Therefore, one does not need to provide the cluster number for hierarchical clustering methods but it is necessary for the partitioning clustering methods.

Hierarchical methods include agglomerative hierarchical methods and divisive hierarchical methods.

The agglomerative hierarchical methods use a bottom-up strategy by treating each individual gene as a cluster at the first step. Then two nearest genes are found and assigned into a cluster where the *nearest* is defined by the distance between these two genes, e.g., for a Pearson distance nearest means the two genes having the largest correlation coefficient. Then an agglomerative hierarchical method assigns a new expression profile for the formed clusters, and repeats these steps until there is only one cluster left.

The divisive hierarchical methods, on the other hand, treat all genes belonging to one cluster at the beginning. Then in each step they choose a partitioning method to divide all genes into a predecided number of clusters, e.g., using k -means to partition genes into two clusters at each single

step. Therefore, the decisive hierarchical clustering methods employ the bottom-down strategy.

The k -means clustering is the simplest and fastest clustering algorithm [42] among the partitioning methods. It has been widely used in many microarray analyses. To form K clusters, the k -means algorithm allocates the observations into different groups in order to minimize the within-group sum of squares

$$\min_{S_k} \left[\sum_{k=1}^K \sum_{i \in S_k} \sum_{j=1}^m (y_{ij} - \bar{y}_{kj})^2 \right],$$

where K is the prespecified cluster number, S_k is the set of objects in the k -th cluster, and \bar{y}_{kj} is the mean of group j in cluster k . In other words, k -means clustering uses the Euclidean distance.

The k -means clusters are formed through iterations as follows: First, k center genes are randomly selected, and every other gene is assigned to the closest center gene. Then, the center is redefined for each cluster to minimize the sum of squares toward the center. In fact, the coordinates of a cluster center are the mean expressions of all the genes in that cluster. After the centers are redefined, all genes are regrouped and the iteration process continues until it converges.

After analyzing a yeast cell cycle expression dataset, *Duan* and *Zhang* [43] noted that it could be particularly useful to use a weighted sum of squares for gene clustering to take into account the loss of synchrony of cells. We refer to *Duan* and *Zhang* [43] for the details.

Another widely used partitioning clustering algorithm is *self-organizing maps* (SOMs) which were developed by *Kohonen* [44]. In essence, SOM clustering is a spatial version of the k -means clustering. For a prespecified grid (i.e., a 6×8 hexagonal grid), SOMs project high-dimensional gene expression data onto a two- or three-dimensional map and place similar genes close to each other. Here, the centroid positions of clusters are related to one another via a spatial topology (e.g., the squared map), and are also iteratively adjusted according to the data.

Both the k -means and SOMs are algorithmic methods and do not have a probabilistic justification. *Probabilistic model-based* clustering (PMC) analysis, on the other hand, assumes that the data is generated by a mixture of underlying probability distributions and uses the maximum likelihood method to estimate parameters that define the number of clusters as well as the clusters. Hence, we do not need to specify the number of clusters. Using the probabilistic model, we can even consider covariates while determining the clustering memberships of the genes. However, the model can quickly become complicated as the number of clusters increases. Thus, we must try to use parsimonious models as

much as possible. Finally, PMC and k -means are also closely related. In fact, k -means can be interpreted as a parsimonious model of simple independent Gaussians [15, 45, 46].

Example 2: Clustering Analysis

In this example, first we perform a hierarchical clustering analysis on the 490 SDE genes from example 1. The clustering analysis is applied in two directions: clustering on samples and clustering on genes. Although we do not present the entire clustering tree here, two major clusters are formed to distinguish tumor and normal samples. For clustering on the genes, there are roughly five major patterns in terms of the gene expressions. One pattern corresponds to the

underexpressed genes and the other four corresponds to the overexpressed genes in the tumor samples versus the normal ones.

For illustration, we selected the first 10 normal arrays and the first 10 cancer arrays, and 20 overexpressed and 20 underexpressed genes randomly from the 490 SDE genes. Figure 32.3 is from the heatmap function in *R*. Though not perfect, two patterns are formed mostly along the line of normal versus tumor tissues. There are roughly five major patterns in terms of expression profiles. Overexpressed and underexpressed genes tend to belong to different clusters. For example, pattern 3 (P3) and pattern 4 (P4) are mainly composed of underexpressed genes,

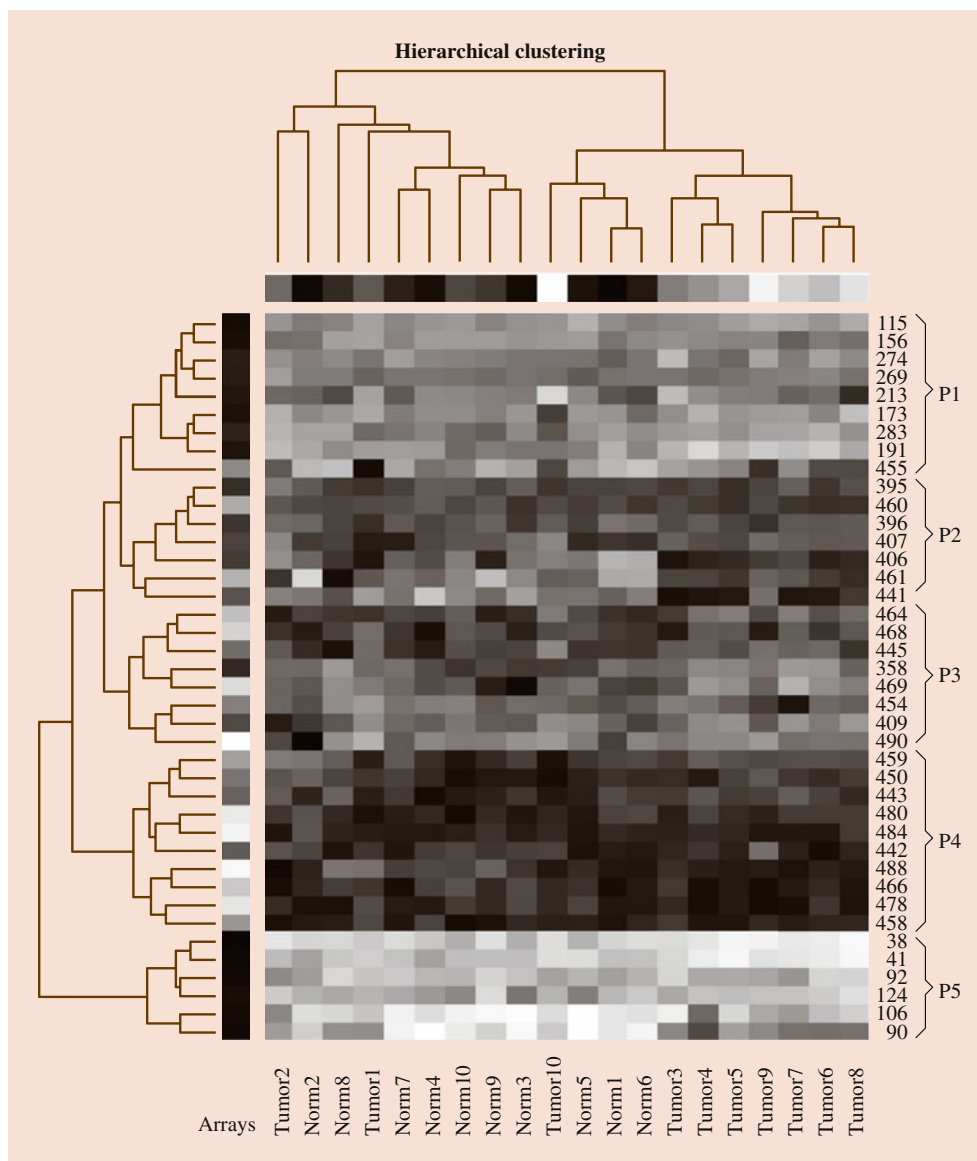


Fig. 32.3 Hierarchical clustering based on a subset of the colon cancer dataset. Each column corresponds to a sample, and each row a gene. The underexpressed genes were assigned numbers above 440, and the overexpressed genes at or below 440

Table 32.1 The numbers of genes belonging to the intersects of the five k -means clusters and the 13 PMC clusters

k -Means clusters	PMC clusters												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	11	0	0	0	57	35	0	0	0	0	29	0
2	25	8	0	0	62	5	0	0	0	0	0	0	0
3	0	0	2	13	0	0	0	0	23	0	0	0	0
4	0	0	15	0	0	0	0	0	1	31	0	0	41
5	0	2	0	0	0	0	1	65	0	34	24	6	0

while the other three clusters contain mainly overexpressed genes.

Following the hierarchical clustering analysis presented above, we also applied the k -means approach to the 490 SDE genes and set the number of clusters to five. Furthermore, we applied *probabilistic model-based* clustering (PMC) to the same dataset. We examined the BIC (Bayesian information criterion) for different numbers of clusters, and it turned out that the value of BIC reaches its minimum at 13 clusters, which is much more than heuristic choice of five. Table 32.1 displays the numbers of genes belonging to the intersects of the five k -means clusters and the 13 PMC clusters. Each of the five k -means clusters is a union of four or so PMC clusters. In fact, if we choose five PMC clusters, they are very similar to the formation of the five k -means clusters, and we will assess this similarity in the next section.

Measure of Agreement Between Two Sets of Clusters

From both the methodological and biologic points of view, there is a need to compare the clusters from different clustering methods. For example, to evaluate the performance of a new clustering approach, we need to compare the derived clusters with the underlying membership in a simulation study. We may also be interested in comparing clustering results derived from the same mRNA samples but being hybridized and analyzed in two different laboratories.

A commonly used measure of agreement between two sets of clusters is the so-called adjusted Rand index (ARI) [15, 47, 48]. Let us consider the partitions U and V , and let n_{ij} be the number of genes falling in the intersect of the i -th cluster in U and the j -th cluster in V . The ARI is defined as

$$\frac{\sum_{i,j} \binom{n_{ij}}{2} - \left[\sum_i \binom{n_i}{2} \sum_j \binom{n_j}{2} \right] / \binom{n}{2}}{\frac{1}{2} \left[\sum_i \binom{n_i}{2} + \sum_j \binom{n_j}{2} \right] - \left[\sum_i \binom{n_i}{2} \sum_j \binom{n_j}{2} \right] / \binom{n}{2}},$$

where n_i and n_j are the numbers of genes in the i -th cluster of U and the j -th cluster of V , respectively.

We suggested some similarity between the k -means and PMC clusters. In fact, the ARI value between the two sets of clusters is 0.425, and it increases to 0.94 if both methods

use five clusters. This similarity is expected, because PMC and k -means are equivalent if PMC assumes an independent Gaussian covariance structure [15].

32.3.2 Classification

In most microarray experiments, we know the groups on the arrays. For example, some mRNA samples were extracted from tumor cells and the others from normal cells. This is similar to the situation in Sect. 32.2.1. Therefore, it is natural to use this information in analysis and to class cells based on the expression profiles. This is so-called supervised learning.

In Sect. 32.2.1, $Y_{ij,k}$ denotes the expression level of the i -th gene on the j -th array in the k -th sample. Here, we also use $(Y_{ij}, Z = k)$ to reflect the fact that the expression level Y_{ij} of the i -th gene on the j -th array comes from the k -th sample. In other words, the sample group is represented by Z , which is the response or dependent variable in classification.

The essence of classification is to define domains in the feature space spanned by Y_{ij} and to assign a class membership Z to each domain. Classification methods differ in the choice of the shape for the domain and in the algorithm to identify the domain. Some elementary concepts are useful to distinguish these differences. The first one is *linearity*. It refers to a linear combination of the features (expressions of different genes) that forms a hyperplane separating different domains in the feature space. The second term is *separability*. It reflects the extent that the different classes of samples are separable. The third concept is *misclassification*. Often, data are only partially separable, and misclassification is inevitable. In this circumstance, we may need to define a cost function to accommodate different classification errors.

In the machine learning literature, there is also a distinction between the *learning* (i.e., training) and the *test* samples. The learning data are used to train the classification algorithm and the test data are used to test the predictive ability of the trained classification algorithm. In practice, however, we usually have one dataset and have to split the sample into the training and test samples by leaving a portion of data out during the learning process and saving it as the test data. This procedure is called cross-validation. More precisely, for a ν -fold cross-validation, we first divide the data into ν approximately equal subsamples. Then, we use $\nu - 1$ subsamples as the training data to construct a classification rule and the leftover subsample as the test data to validate the classification rule. After rotating every subsample between training and test data, the performance of the classification rule is assessed through the average in the ν runs of validation in the test sample.

In the next subsections, we will review four classification methods that are useful for classifying tissue samples based on gene expression profiles. The methods are linear discrim-

inant analysis (LDA), support vector machines (SVM), artificial neural networks (ANN), and tree-based classification.

LDA

LDA was introduced by Fisher in 1936 for classifying samples by finding a hyperplane that maximizes the between-class variances. Let S_Y be the common sample covariance matrix of all gene expressions, \bar{Y}_1 and \bar{Y}_2 be the average expression levels of the genes in groups 1 and 2, respectively. The solution to LDA is $S_Y^{-1}(\bar{Y}_1 - \bar{Y}_2)$.

SVM

SVM was first proposed by *Boser et al.* [49] and *Cortes and Vapnik* [50]. SVM finds an optimal hyperplane to separate samples and to allow the maximum *separation* between different classes of samples. The margin of the region that separates samples is *supported* by a few vectors, termed support vectors.

In a two-class classification problem, let $Z = 1$ or -1 denote the two classes. If the two classes of samples are separable, we find a hyperplane $\{y : y^T \beta + \beta_0 = 0, \|\beta\| = 1\}$ such that $(y^T \beta + \beta_0)Z \geq C \geq 0$, where C is the margin optimized to allow the maximal space between the two classes of samples.

For nonseparable case, the procedure is much complicated. Some points will inevitably be on the wrong side of the hyperplane. The idea is to introduce a slack variable to reflect how far a sample is on the wrong side, and then look for the hyperplane at the condition of the total misclassification less than a user-selected limit (i.e., bound the sum of slack variables by a constant). We refer to *Vapnik* [51] for the details.

Example 3: Support Vector Machine (SVM)

In this example, we perform a classification analysis on the colon cancer data by SVM. We use M26697 and M63391, the two most significant genes that were identified by SAM from example 1. Specifically, M26697 is the most significant overexpressed gene and M63391 is the most significant underexpressed gene. We used the SVM function in *R* with the cost equal to 100, γ of 1 and tenfold cross-validation, where γ is the coefficient of the radial kernel used to form a hyperplane. Figure 32.4 displays the contour plot of the SVM result. The prediction model correctly classifies 37 cancer and 20 normal samples, but misclassifies three cancer and two normal samples.

Neural Network

The artificial neural network (ANN) is a very popular methodology in machine learning. Also referred to as connectionist architectures, parallel distributed processing, and neuromorphic systems, ANN is an information-processing paradigm with collections of mathematical models that emulate the densely interconnected, parallel

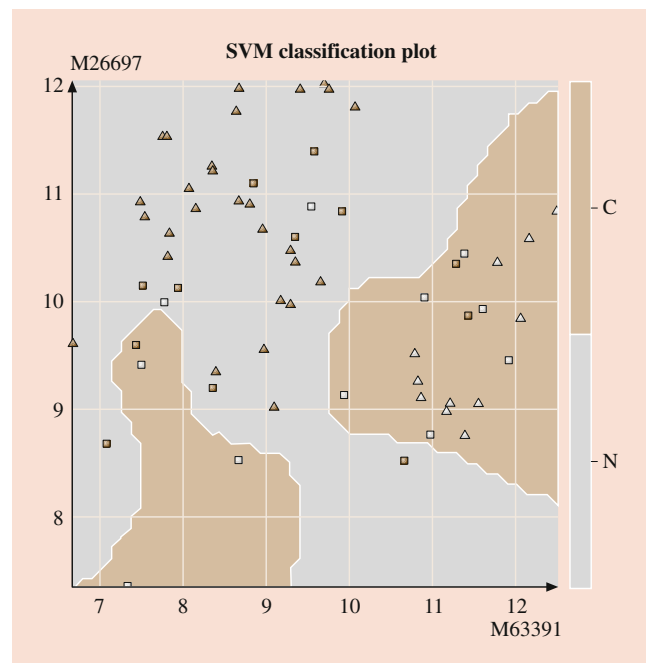


Fig. 32.4 Contour plot of the SVM result using two genes: M26997 and M63391 for the colon cancer data. C represents cancer and N represents normal. The *light-gray area* is the cancer region and the *brown area* is the normal region. *Square points* represent the support vectors and the *triangle points* represent the data points other than support vectors. The *brown* and *white points* belong to the cancer and the normal regions, respectively

structure of the mammalian brain and adaptive biological learning. It is composed of a large number of highly interconnected processing elements that are analogous to neurons and are tied together with weighted connections that are analogous to synapses. Learning typically occurs by example through training, or exposure to a true set of input/output data where the training algorithm iteratively adjusts the connection weights (synapses). These connection weights store the knowledge necessary to solve specific problems.

ANN can be used for feature selection and feature extraction. The former amounts to variable selection and reduction in statistics and the latter is a generation of the statistical techniques such as principal component analysis, factor analysis, and linear discriminant analysis that are intended to identify lower-dimensional data structures such as linear directions. These lower-dimensional structures usually depend on all of the original variables (i.e., features). Thus, ANN is in essence a computationally intensive version of traditional statistical methods such as regression, classification, clustering, and factor analysis. However, ANN is designed in a way that mimics neural networks and is biologically intuitive and appealing in many applications. This is the major reason that we plan to consider ANN

as one of the primary tools to explore the unknown relationship in our data, which is usually referred to as pattern recognition.

The advantage of ANNs lies in their resilience against distortions in the input data and their capability for learning. They are often good at solving problems that are too complex for conventional technologies (e.g., problems that do not have an algorithmic solution, or for which an algorithmic solution is too complex to be found), and are often well suited to problems that people are good at solving, but for which traditional methods are not.

There are multitudes of different types of ANNs. Some of the more popular include the multilayer perceptron, which is generally trained with the backpropagation of error algorithm, learning vector quantization, radial basis functions, Hopfield, and Kohonen, to name a few. Some ANNs are classified as feed-forward while others are recurrent (i.e., implement feedback) depending on how data is processed through the network. Some ANNs employ supervised training while others are referred to as unsupervised or self-organizing.

Figure 32.5 illustrates a conventional three-layer neural network with n features and K classes. For this feed-forward neural network, the inputs are y_1, \dots, y_n which correspond to the gene expression profiles and the outputs are z_1, \dots, z_K , which correspond to the K samples in the microarray data. The middle layer consists of many hidden units (also called neurons) and the number of hidden units can be freely chosen and determine the maximum nonlinearity. Each line in Fig. 32.5 indicates a weight – the edge – in the network. This weight represents how much the two neurons which are connected by it can interact. If the weight is larger, then the two neurons can interact more, that is, a stronger signal can

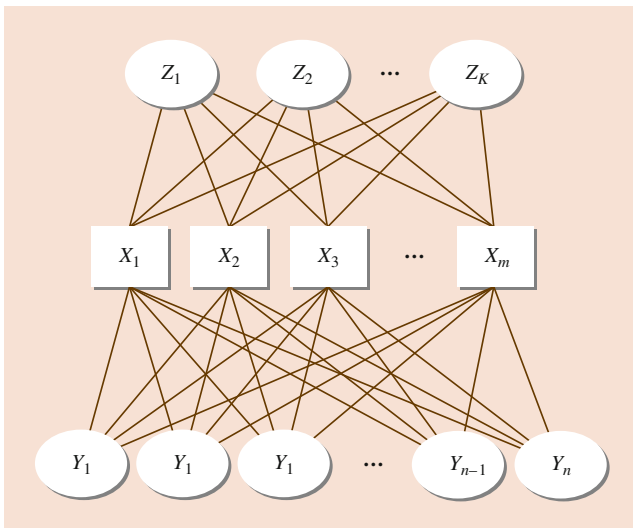


Fig. 32.5 Architecture of a conventional three-layered feed-forward neural network

pass through the edge. The nature of the interconnections between two neurons can be such that one neuron can either stimulate (a positive weight α) or inhibit (a negative weight α) the other. More precisely, in each hidden unit, we have

$$X_m = \sigma(\alpha_{0m} + \alpha_m^T Y),$$

where σ is called the activation function or neural function and $(\alpha_{0m}, \alpha_m^T)$ are the weights. A common choice for σ is the sigmoid function

$$\sigma(v) = \frac{1}{1 + e^{-v}}.$$

The output function allows a final transformation of the linear combinations of the hidden unit variables

$$f_k(z) = g_k(\beta_{0k} + \beta_k^T X).$$

For a K -class classification, a softmax (logistic) function is usually chosen for the output function

$$g_k(T) = \frac{e^{T_k}}{\sum_{l=1}^K e^{T_l}}.$$

During the training period, we present the perceptron with inputs one at a time and see what output it gives. If the output is wrong, we will tell it that it has made a mistake. It should then change its weights and/or threshold properly to avoid making the same mistake later.

32.3.3 Tree- and Forest-Based Classification

One of the most convenient and intuitive approaches for classification is classification trees [52–54]. Classification trees, and their expansion to forests, are based on the so-called recursive partitioning technique. The basic idea of recursive partitioning is to extract homogeneous strata of the tissue samples through expression profiles depending on the expression levels of a particular gene.

Zhang and Yu [55] reanalyzed the dataset from Hedenfalk et al. [56] to classify breast cancer mutations in either the BRCA1 or BRCA2 gene using gene expression profiles. Hedenfalk et al. [56] collected and analyzed biopsy specimens of primary breast cancer tumors from seven and eight patients with germline mutations of BRCA1 and BRCA2, respectively. In addition, seven patients with sporadic cases of primary breast cancer whose family history was unknown were also identified. They obtained cDNA microarrays from 5361 unique genes, of which 2905 are known genes and 2456

are unknown. Thus, in this dataset, let $Z = 1, 2, 3$ denote BRCA1, BRCA2, and sporadic cases, respectively.

If we use this entire breast cancer dataset to construct a tree, these 22 samples form the initial learning sample, which is called the root node and labeled as node 1 in the tree diagram (Fig. 32.6). The tree structure is determined by recursively selecting a split to divide an upper layer node into two offspring nodes. To do this, we need to evaluate the homogeneity, or the impurity to its opposite, of any node. A common measure of node impurity is the entropy function

$$i_t = - \sum_{k=1}^K P(Z = k | \text{node } t) \log [P(Z = k | \text{node } t)].$$

If node t is the root node, then $P(Z = 1 | \text{node } t) = 7/22$, $P(Z = 2 | \text{node } t) = 8/22$, and $P(Z = 3 | \text{node } t) = 7/22$. Thus, the impurity i_t of the root node can be calculated easily as follows: $i_t = -(7/22) \log(7/22) - (8/22) \log(8/22) - (7/22) \log(7/22) = 1.097$.

How good is the root node? The impurity is zero for a perfect node in which $P(Z = k | \text{Node } t)$ is either 0 or 1, and reaches its worst level when $P(Z = k | \text{node } t) = \frac{1}{3}$ with $i_t = 1.099$. Therefore, the impurity of the root node is near the worst level by design, motivating us to partition the root node into small nodes to reduce the impurity.

The first step of the recursive partitioning process is to divide the root of 32 samples in Fig. 32.6 into two nodes, namely, nodes 2 and 3 in Fig. 32.6. There are many ways of partitioning the root node, because we can take any of the 5361 genes and split the root node according to whether the expression level of this chosen gene is greater than any threshold c . After comparing all possible partitions, we choose the gene and its threshold to keep both i_2 in node 2 and i_3 in node 3 at their lowest possible levels simultaneously. Mathematically, we achieve this goal by minimizing the

weighted impurity $r_2 i_2 + r_3 i_3$, where r_2 and r_3 are the proportions of tissue samples in nodes 2 and 3, respectively. This is precisely how the first split (i.e., whether $ST13 > 0.835$) in Fig. 32.6 is determined.

Once the root is split into nodes 2 and 3, and we can apply the same procedure to potentially split nodes 2 and 3 further. Indeed, the tree in Fig. 32.6 divides the 22 samples into four groups using Heping Zhang’s RTREE (<http://c2s2.yale.edu/software>). Nodes 2 and 3 are divided based on the expression levels of genes ARF3 and LRBA.

Using a variety of analytic techniques including a modified F- and t -test and a mutual information scoring, Hedenfalk et al. [56] selected nine differentially expressed genes to classify BRCA1-mutation-positive and negative tumors and then 11 genes for BRCA2-mutation-positive and negative tumors. Clearly, the tree in Fig. 32.6 uses fewer genes and is a much simpler classification rule.

Although Fig. 32.6 is simple, it does not contain the potentially rich information in the dataset. To improve the reliability of the classification and to accommodate potentially multiple biological pathways, Zhang and Yu [55] and Zhang et al. [57] proposed expanding trees to forests. The large number of genes in microarray makes it an ideal application for these forests.

The most common approach to constructing forests is to perturb the data randomly, form a tree from the perturbed data, and repeat this process to form a series of trees; this is called a random forest. After a forest is formed, we aggregate information from the forest. One such scheme, called bagging (bootstrapping and aggregating), generates a bootstrap sample from the original sample. The final classification is then based on the majority vote of all trees in the forest [58].

It is well known that random forests [18, 58] improve predictive power in classification. After observing the fact that there are typically many trees that are of equally high predictive quality in analyzing genomic data, Zhang et al. [18] proposed a method to construct forests in a deterministic manner. Deterministic forests eliminate the randomness in the random forests and maintain a similar, and sometimes improved, level of precision as the random forests.

The procedure for constructing the deterministic forests is simple. We can search and collect all distinct trees that have a nearly perfect classification or are better than any specified precision. This can be carried out by ranking the trees in deterministic forests. One limitation for the forests (random or deterministic) is that we cannot view all trees in the forests. However, we can examine the frequency of genes as they appear in the forests. Frequent and prominent genes may then be used and analyzed by any method as described above. In other words, forest construction offers a mechanism for data reduction. For the breast cancer data, one of the most

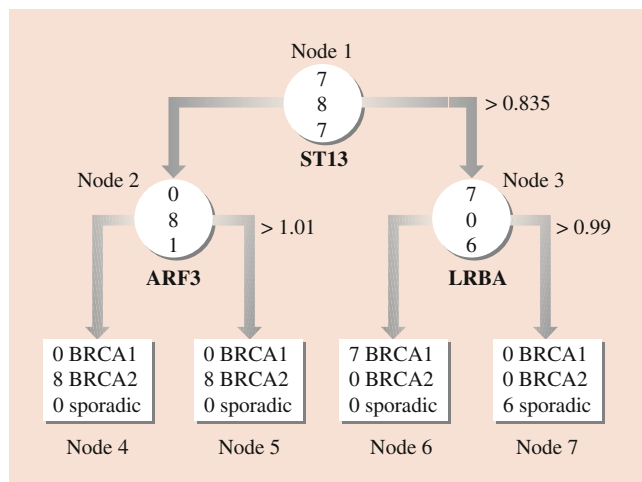


Fig. 32.6 Classification tree for breast cancer data

prominent genes identified in the forests is ERBB2. *Kroll et al.* [59] analyzed the gene expression patterns of four breast cancer cell lines: MCF-7, SK-BR-3, T-47D, and BT-474, and reported unique high levels of expressions in the receptor tyrosine kinase ERBB2.

32.4 Fourth-Level Analysis of Microarray Data

Nowadays, different types and platforms of microarray have been developed to address the same (or similar) biological problems. How to integrate and exchange the information contained in different sources of studies effectively is an important and challenging topic for both biologists and statisticians [60]. The strategy depends on the situation. When all studies of interest were conducted under the same experimental conditions, this is a standard situation for meta-analysis. There are situations where the experiments are similar, but different platforms were measured, such as the integration of one cDNA array-based study and one oligonucleotide array-based study. There are also situations where different biomolecule microarrays were collected, such as the integration of a genomic array study and a proteomic array study.

Integrating a cDNA array and an Affymetrix chip is complicated because genes on a cDNA array may correspond to several genes (or probesets) on the Affymetrix chip based on the Unigene cluster-matching criteria [61]. Instead of matching by genes, matching by the sequence-verified probes may increase the correlation between two studies [62].

Most meta-analyses of microarray data have been performed in a study-by-study manner. For example, *Yauk et al.* [63] use the Pearson coefficient to measure the correlation across studies; *Rhodes et al.* [64] and *Wang et al.* [65] use the estimations from one study as prior knowledge while analyzing other studies; and *Welsh et al.* [66] treat DNA microarrays as a screening tool and then use protein microarrays to identify the biomarker in cancer research. While they are convenient, these strategies are not ideal [64, 67]. Thus, it is imperative and useful to develop better methods to synthesize information from different genomic and proteomic studies [60, 63, 68].

32.5 Final Remarks

The technology of gene and protein chips is advancing rapidly, and the entire human genome can be simultaneously monitored on a single chip. The analytic methodology

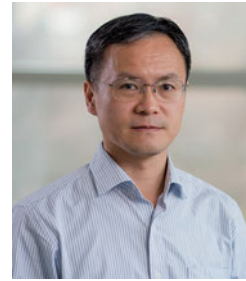
is evolving together with the technology development, but is far from satisfactory. This article reviews some of the commonly used methods in analyzing microarrays. Analyzing microarray data is still challenging; some of the important issues include how to interpret the results in the biological context, how to improve the reproducibility of the conclusions, and how to integrate information from related but different studies.

References

1. Schena, M., Shalon, M., Davis, R.W., Brown, P.O.: Quantitative monitoring of gene-expression patterns with a complementary-DNA microarray. *Science*. **270**, 467–470 (1995)
2. Heller, R.A., Schena, M., Chai, A., Shalon, D., Bedilion, T., Gilmore, J., Woolley, D.E., Davis, R.W.: Discovery and analysis of inflammatory disease-related genes using cDNA microarrays. *Proc. Natl. Acad. Sci. U. S. A.* **94**(6), 2150–2155 (1997)
3. Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D., Friedman, N.: Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat. Genet.* **34**, 166–176 (2003)
4. Hacia, J.C., Sun, B., Hunt, N., Edgemon, K., Mosbrook, D., Robbins, C., Fodor, S.P.A., Tagle, D.A., Collins, F.S.: Strategies for mutational analysis of the large multiexon ATM gene using high-density oligonucleotide arrays. *Genome Res.* **8**, 1245–1258 (1998)
5. Fan, J.B., Chen, X.Q., Halushka, M.K., Berno, A., Huang, X.H., Ryder, T., Lipshutz, R.J., Lockhart, D.J., Chakravarti, A.: Parallel genotyping of human SNPs using generic high-density oligonucleotide tag arrays. *Gen. Res.* **10**, 853–860 (2000)
6. Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C.H., Angelo, M., Ladd, C., Reich, M., Latulippe, E., Mesirov, J.P., Poggio, T., Gerald, W., Loda, M., Lander, E.S., Golub, T.R.: Multiclass cancer diagnosis using tumor gene expression signatures. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 15149–15154 (2001)
7. Marcotte, E.R., Srivastava, L.K., Quirion, R.: DNA microarrays in neuropsychopharmacology. *Trends Pharmacol. Sci.* **22**, 426–436 (2001)
8. Li, C., Wong, W.H.: Model-based analysis of oligonucleotide arrays: expression index computation, outlier detection. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 31–36 (2001)
9. Efron, B., Tibshirani, R., Storey, J.D., Tusher, V.: *J. Amer. Stat. Assoc.* **96**, 1151–1160 (2001)
10. Tusher, V.G., Tibshirani, R., Chu, G.: Significance analysis of microarrays applied to the ionizing radiation response. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 5116–5121 (2001)
11. Irizarry, R.A., Hobbs, B., Collin, F., Beazer-Barclay, Y.D., Antonellis, K.J., Scherf, U., Speed, T.P.: Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics.* **4**, 249–264 (2003)
12. Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D.: Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. U. S. A.* **95**, 14863–14868 (1998)
13. Soukas, A., Cohen, P., Succi, N.D., Friedman, J.M.: Leptin-specific patterns of gene expression in white adipose tissue. *Genes Dev.* **14**(8), 963–980 (2000)
14. Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E.S., Golub, T.R.: Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proc. Natl. Acad. Sci. U. S. A.* **96**(6), 2907–2912 (1999)

15. Yeung, K.Y., Ruzzo, W.L.: Principal component analysis for clustering gene expression data. *Bioinformatics*. **17**, 763–774 (2001)
16. Yeung, K.Y., Fraley, C., Murua, A., Raftery, A.E., Ruzzo, W.L.: Model-based clustering and data transformations for gene expression data. *Bioinformatics*. **17**, 977–987 (2001)
17. Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., Altman, R.B.: Missing value estimation methods for DNA microarrays. *Bioinformatics*. **17**(6), 520–525 (2001)
18. Zhang, H.P., Yu, C., Singer, B.: Cell and tumor classification using gene expression data: construction of forests. *Proc. Natl. Acad. Sci. U. S. A.* **100**, 4168–4172 (2003)
19. Furey, T.S., Cristianini, N., Duffy, N., Bednarski, D.W., Schummer, M., Haussler, D.: Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*. **16**(10), 906–914 (2000)
20. Mehrotra, K., Mohan, C.K., Ranka, S.: *Elements of Artificial Neural Networks*. MIT, Massachusetts (1997)
21. Zhang, H.P., Yu, C., Singer, B., Xiong, M.: Recursive partitioning for tumor classification with gene expression microarray data. *Proc. Natl. Acad. Sci. U. S. A.* **98**, 6730–6735 (2001)
22. Butte, A.J., Tamayo, P., Slonim, D., Golub, T.R., Kohane, I.S.: Discovering functional relationships between RNA expression and chemotherapeutic susceptibility using relevance networks. *Proc. Natl. Acad. Sci. U. S. A.* **97**, 12182–12186 (2000)
23. D'haeseleer, P., Liang, S., Somogyi, R.: *Gene Expression Data Analysis and Modeling*. Pacific Symposium on Biocomputing (1999)
24. Shmulevich, I., Dougherty, E.R., Kim, S., Zhang, W.: Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*. **18**(2), 261–274 (2002)
25. Friedman, N., Lital, M., Nachman, I., Pe'er, D.: Using Bayesian networks to analyze expression data. *J. Comp. Biol.* **7**, 601–620 (2000)
26. Segal, E., Taskar, B., Gasch, A., Friedman, N., Koller, D.: Rich probabilistic models for gene expression. *Bioinformatics*. **1**, 1–10 (2001)
27. Lockhart, D.J., Dong, H., Byrne, M.C., Folletti, M.T., Gallo, M.V., Chee, M.S., Mittmann, M., Wang, C., Kobayashi, M., Horton, H., Brown, E.L.: Expression monitoring by hybridization to high-density oligonucleotide arrays. *Nat. Biotechnol.* **14**, 1675–1680 (1996)
28. Smyth, G.: Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Stat. Appl. Genet. Mol. Biol.* **3**(1), 3 (2004)
29. Šidák, Z.: Rectangular confidence regions for the means of multivariate normal distributions. *J. Am. Stat. Assoc.* **62**, 626–633 (1967)
30. Draghici, S.: *Data Analysis Tools for DNA Microarrays*. Chapman & Hall/CRC, New York (2003)
31. Benjamin, Y., Hochberg, Y.: Controlling the false discovery rate – a practical and powerful approach to multiple testing. *J. Roy. Soc. B Met.* **57**(1), 289–300 (1995)
32. Storey, J.D.: A direct approach to false discovery rates. *J. R. Stat. Ser. B Stat. Methodol.* **64**, 479–498. Part 3 (2002)
33. Storey, J.D.: A Bayesian interpretation, the q-value. *Ann. Stat.* **31**(6), 2013–2035 (2003)
34. Troendle, J.F.: Stepwise normal theory multiple test procedures controlling the false discovery rate. *J. Stat. Plan. Inference.* **84**(1–2), 139–158 (2000)
35. Efron, B., Tibshirani, R.: Empirical bayes methods and false discovery rates for microarrays. *Genet. Epidemiol.* **23**(1), 70–86 (2002)
36. Lonnstedt, I., Speed, T.: Replicated microarray data. *Stat. Sinica.* **12**(1), 31–46 (2001)
37. Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., Levine, A.J.: Broad patterns of gene expression revealed by clustering analysis of tumor, normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. U. S. A.* **96**, 6745–6750 (1999)
38. Quackenbush, J.: Computational analysis of microarray analysis. *Nat. Rev. Genet.* **2**, 418–427 (2001)
39. Kaminski, N., Friedman, N.: Practical approaches to analyzing results of microarray experiments. *Am. J. Respir. Cell Mol. Biol.* **27**(2), 125–132 (2002)
40. Jansen, R., Greenbaum, D., Gerstein, M.: Relating whole-genome expression data with protein-protein interactions. *Genome Res.* **12**(1), 37–46 (2002)
41. Boldrick, J.C., Alizadeh, A.A., Diehn, M., Dudoit, S., Liu, C.L., Belcher, C.E., Botstein, D., Staudt, L.M., Brown, P.O., Relman, D.A.: Stereotyped and specific gene expression programs in human innate immune responses to bacteria. *Proc. Natl. Acad. Sci. U. S. A.* **99**, 972–977 (2002)
42. Sherlock, G.: Analysis of large-scale gene expression data. *Curr. Opin. Immunol.* **12**(2), 201–205 (2000)
43. Duan, F.H., Zhang, H.P.: Correcting the loss of cell-cycle synchrony in clustering analysis of microarray data using weights. *Bioinformatics*. **20**(11), 1766–1771 (2004)
44. Kohonen, T.: *Self-Organizing Maps*. Springer, Berlin Heidelberg New York (1997)
45. Venables, W.N., Ripley, B.D.: *Modern Applied Statistics with S*. Springer, Berlin Heidelberg New York (2002)
46. Wit, E., McClure, J.: *Statistics for Microarrays*. Wiley, New York (2004)
47. Hubert, L., Arabie, P.: Comparing partitions. *J. Classif.* **2**, 193–218 (1985)
48. Milligan, G.W., Cooper, M.C.: A study of the comparability of external criteria for hierarchical cluster-analysis. *Multivar. Behav. Res.* **21**(4), 441–458 (1986)
49. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Haussler, D. (ed.) *Fifth Annual Workshop on Computational Learning Theory*, pp. 144–152. ACM, New York (1992)
50. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)
51. Vapnik, V.: *Statistical Learning Theory*. Wiley, New York (1998)
52. Breiman, L., Friedman, J., Stone, C., Olshen, R.: *Classification, Regression Trees*. Wadsworth, Belmont (1984)
53. Zhang, H.P., Singer, B.: *Recursive Partitioning in the Health Sciences*. Springer, Berlin Heidelberg New York (1999)
54. Zhang, H.P., Singer, B.: *Recursive Partitioning and Applications*. Springer, New York (2010)
55. Zhang, H., Yu, C.-Y.: Tree-based analysis of microarray data for classifying breast cancer. *Front. Biosci.* **7**, c63–c67 (2002)
56. Hedenfalk, I., Duggan, D., Chen, Y., Radmacher, M., Bittner, M., Simon, R., Meltzer, P., Gusterson, B., Esteller, M., Raffeld, M., Yakhini, Z., Ben-Dor, A., Dougherty, E., Kononen, J., Bubendorf, L., Fehrl, W., Pittaluga, S., Gruvberger, S., Loman, N., Johannsson, O., Olsson, H., Wilfond, B., Sauter, G., Kallioniemi, O.P., Borg, A., Trent, J.: Gene-expression profiles in hereditary breast cancer. *N. Engl. J. Med.* **344**, 539–548 (2001)
57. Zhang, H.P., Yu, C.Y., Zhu, H.T., Shi, J.: Identification of linear directions in multivariate adaptive spline models. *J. Am. Stat. Assoc.* **98**, 369–376 (2003)
58. Random, B.L.: Random forests. *Mach. Learn.* **45**, 5–32 (2001)

59. Kroll, T., Odyvanova, L., Clement, H., Platzer, C., Naumann, A., Marr, N., Hoffken, K., Wolfl, S.: Molecular characterization of breast cancer cell lines by expression profiling. *J. Cancer Res. Clin. Oncol.* **128**, 125–134 (2002)
60. Moreau, Y., Aerts, S., Moor, B.D., Strooper, B.D., Dabrowski, M.: Comparison and meta-analysis of microarray data: from the bench to the computer desk. *Trends Genet.* **9**(10), 570–577 (2003)
61. Ghosh, D., Barette, T., Rhodes, D., Chinnaiyan, A.: Statistical issues and methods for meta-analysis of microarray data: a case study in prostate cancer. *Funct. Integrat. Gen.* **3**(4), 180–188 (2003)
62. Mecham, B.H., Klus, G.T., Strover, J., Augustus, M., Byrne, D., Bozso, P., Wetmore, D.Z., Mariani, T.J., Kohane, I.S., Szallasi, Z.: Sequence-matched robes produce increased cross-platform consistency and more reproducible biological results in microarray-based gene expression measurements. *Nucleotide Acids Res.* **32**(9), e74 (2004)
63. Yauk, C.L., Berndt, M.L., Williams, A., Douglas, G.R.: Comprehensive comparison of six microarray technologies. *Nucleic Acids Res.* **32**(15), e124 (2004)
64. Rhodes, D.R., Barrette, T.R., Rubin, M.A., Ghosh, D., Chinnaiyan, A.M.: Meta-analysis of microarrays: interstudy validation of gene expression profiles reveals pathway dysregulation in prostate cancer. *Cancer Res.* **62**(15), 4427–4433 (2002)
65. Wang, J., Coombes, K.R., Highsmith, W.E., Keating, M.J., Abruzzo, L.V.: Differences in gene expression between B-cell chronic lymphocytic leukemia and normal B cells. *Bioinformatics.* **20**(17), 3166–3178 (2004)
66. Welsh, J.B., Sapinoso, L.M., Kern, S.G., Brown, D.A., Liu, T., Bauskin, A.R., Ward, R.L., Hawkins, N.J., Quinn, D.I., Russell, P.J., Sutherland, R.L., Breit, S.N., Moskaluk, C.A., Frierson Jr., H.F., Hampton, G.M.: Large-scale delineation of secreted protein biomarkers overexpressed in cancer tissue and serum. *Proc. Natl. Acad. Sci.* **100**(6), 3410–3415 (2003)
67. Hedges, L.V., Olkin, I.: *Statistical Methods for Meta-Analysis*. Academic, New York (1985)
68. Järvinena, A.K., Hautaniemi, S., Edgrena, H., Auvinen, P., Saarela, J., Kallioniemi, O.P., Monni, O.: Are data from different gene expression microarray platforms comparable? *Genomics.* **83**(6), 1164–1168 (2004)



Fenghai Duan was born in Heilongjiang, China. After completing his bachelor's degree in biochemistry at Fudan University in 1995, he received his master's degree in molecular biology from the Institute of Biophysics, Academia Sinica, in 1998. In the year 2000, Duan joined the Ph.D. program in biostatistics at Yale University and worked on his thesis under the supervision of Professor Heping Zhang. Duan's doctoral dissertation was about the analysis of microarray experiments and was awarded the Ph.D. degree in May 2005. Currently, he is an associate professor of biostatistics at Brown University. His research interests are in the development of statistical methods for the analysis of high-dimensional biological data.



Heping Zhang is the Susan Dwight Bliss Professor of Biostatistics, Child Study, and Statistics and Data Science at the Yale University. He is interested in development of statistical methods and software and their applications in biomedical studies, particularly in behavioral science, epidemiology, genetics, psychiatry, and pregnancy outcomes. He publishes extensively on tree- and spline-based methods as well as latent variable models for genetic studies of ordinal traits. He is a fellow of the American Statistical Association and Institute of the Mathematical Statistics, and the editor of the *Journal of the American Statistical Association – Applications and Case Studies*.



Mitsuo Gen and Lin Lin

Contents

33.1 Foundations of Genetic Algorithms	636	33.5.2 Reliability Design with Redundant Unit and Alternatives	661
33.1.1 Basic Genetic Algorithm.....	636	33.5.3 Network Reliability Design	661
33.1.2 Hybrid Genetic Algorithms.....	637	33.5.4 Tree-Based Network Design	661
33.1.3 Adaptive Genetic Algorithms.....	638	33.6 Logistics Network Problems	662
33.1.4 Fuzzy Logic Controller (FLC).....	638	33.6.1 Linear Transportation Model.....	662
33.1.5 Multiobjective Optimization Problems.....	639	33.6.2 Multiobjective Transportation.....	663
33.1.6 Genetic Algorithm with Learning.....	641	33.6.3 Bicriteria Transportation Model with Fuzzy Coefficients.....	663
33.2 Combinatorial Optimization Problems	642	33.6.4 Multistage Logistics Model.....	664
33.2.1 Knapsack Problem.....	643	33.6.5 SCM Network Design.....	665
33.2.2 Minimum Spanning Tree Model.....	643	33.6.6 Reverse Logistics Model.....	666
33.2.3 Set-Covering Model.....	644	33.7 Location and Allocation Problems	667
33.2.4 Bin-Packing Model.....	644	33.7.1 Location-Allocation Model.....	667
33.2.5 Traveling-Salesman Model.....	645	33.7.2 Capacitated Plant Location Model.....	668
33.3 Network Design Problems	646	33.7.3 Obstacle Location-Allocation Model.....	669
33.3.1 Shortest-Path Model.....	646	33.7.4 LAP with Multi-facility Service.....	670
33.3.2 Maximum-Flow Model.....	647	References	672
33.3.3 Minimum-Cost-Flow Model.....	648		
33.3.4 Centralized Network Design.....	648		
33.3.5 Multistage Process Planning Model.....	649		
33.3.6 Bicriteria MXF/MCF Model.....	650		
33.4 Scheduling Problem	652		
33.4.1 Flow-Shop Sequencing Model.....	652		
33.4.2 Job-Shop Scheduling Model.....	653		
33.4.3 Flexible Job-Shop Scheduling Model.....	654		
33.4.4 Resource-Constrained Project Scheduling Model.....	656		
33.4.5 Multiprocessor Scheduling Model.....	656		
33.4.6 Assembly Line Balancing.....	657		
33.5 Reliability Design Problem	659		
33.5.1 Genetic Algorithm for Reliability Optimization.....	660		

Abstract

The first part of this chapter describes the foundation of genetic algorithms. It includes hybrid genetic algorithms, adaptive genetic algorithms, and fuzzy logic controllers. After a short introduction to genetic algorithms, the second part describes combinatorial optimization problems including the knapsack problem, the minimum spanning tree problem, the set-covering problem, the bin-packing problem, and the traveling-salesman problem; these are combinatorial optimization problems which are characterized by a finite number of feasible solutions. The third part describes network design problems. Network design and routing are important issues in the building and expansion of computer networks. In this part, the shortest-path problem, maximum-flow problem, minimum-cost-flow problem, centralized network design, and multistage process planning problem are introduced. These problems are typical network problems and have been studied for a long time. The fourth section describes scheduling problems. Many scheduling problems from

M. Gen (✉)
 Fuzzy Logic Systems Institute (FLSI), Department of Research,
 Fukuoka, Japan
 Tokyo University of Science (TUS), Research Institute of Science &
 Technology, Tokyo, Japan
 e-mail: gen@flsi.or.jp; gen@rs.tus.ac.jp
 L. Lin
 Dalian University of Technology, Economy and Technology
 Development Area, Dalian, China
 e-mail: lin@dlut.edu.cn

manufacturing industries are quite complex in nature and very difficult to solve by conventional optimization techniques. In this part the flow-shop sequencing problem, job-shop scheduling, the resource-constrained project scheduling problem, and multiprocessor scheduling are introduced. The fifth part introduces the reliability design problem, including simple genetic algorithms for reliability optimization, reliability design with redundant units and alternatives, network reliability design, and tree-based network topology design. The sixth part describes logistic problems including the linear transportation problem, the multiobjective transportation problem, the bicriteria transportation problem with fuzzy coefficients, and supply chain management network design. Finally, the last part describes location and allocation problems including the location-allocation problem, capacitated plant-location problem, and obstacle location-allocation problem.

Keywords

Genetic algorithms · Network design · Scheduling · Reliability design · Logistics · Location and allocation

33.1 Foundations of Genetic Algorithms

Recently, genetic algorithms (GAs) have received considerable attention regarding their potential as an optimization technique for complex problems and have been successfully applied in the area of industrial engineering. The well-known applications include scheduling and sequencing, reliability design, vehicle routing location, transportation, and many others.

There are three major advantages when applying GA to optimization problems:

1. **Adaptability:** GA does not have much mathematical requirements about the optimization problems. Due to the evolutionary nature, GA will search for solutions without regard to the specific inner workings of the problem. GA can handle any kind of objective functions and any kind of constraints, i.e., linear or nonlinear, defined on discrete, continuous, or mixed search spaces.
2. **Robustness:** The use of evolution operators makes GA very effective in performing global search (in probability), while most of conventional heuristics usually perform local search (LS). It has been proved by many studies that GA is more efficient and more robust in locating optimal solution and reducing computational effort than other conventional heuristics.
3. **Flexibility:** GA provides us a great flexibility to hybridize with domain-dependent heuristics to make an efficient implementation for a specific problem.

33.1.1 Basic Genetic Algorithm

Since the 1960s there has been being an increasing interest in imitating living beings to solve such kinds of hard optimization problems. Simulating natural evolutionary processes of human beings results in stochastic optimization techniques called evolutionary algorithms (EAs) that can often outperform conventional optimization methods when applied to difficult real-world problems. EAs mostly involve metaheuristic optimization algorithms such as genetic algorithm (GA) [1], evolutionary programming (EP), evolution strategy (ES), genetic programming (GP), particle swarm optimization (PSO), and ant colony optimization (ACO) [2, 3]. Among them, genetic algorithms are perhaps the most widely known type of evolutionary algorithms used today [4].

Genetic algorithms are stochastic search algorithms based on the mechanism of natural selection and natural genetics. Genetic algorithms, in contrast to conventional search techniques, start with an initial set of random solutions called the population. Each individual in the population is called a chromosome, encoding a solution to the problem at hand. A chromosome is a string of symbols, usually, but not necessarily, a binary bit string. The chromosomes *evolve* through successive iterations, called generations. During each generation, the chromosomes are evaluated, using some measures of fitness [5]. To create the next generation, new chromosomes, called offspring, are formed by either merging two chromosomes from the current generation using a *crossover* operator or modifying a chromosome using a mutation operator.

A new generation is formed by selecting, according to the fitness values, some of the parents and offspring and rejecting others so as to keep the population size constant.

Fitter chromosomes have higher probabilities of being selected. After several generations, the algorithms converge to the best chromosome, which we hope represents the optimum or suboptimal solution to the problem when decoded. Let $P(t)$ and $C(t)$ be population or parents and offspring in the current generation t ; the general structure of GAs (Fig. 33.1) is described as follows:

procedure: Genetic Algorithms (GA)

input: problem data, parameters

output: the best solution

begin

$t \leftarrow 0$;

initialize $P(t)$ by encoding routine;

evaluate $P(t)$ by decoding routine & keep best solution;

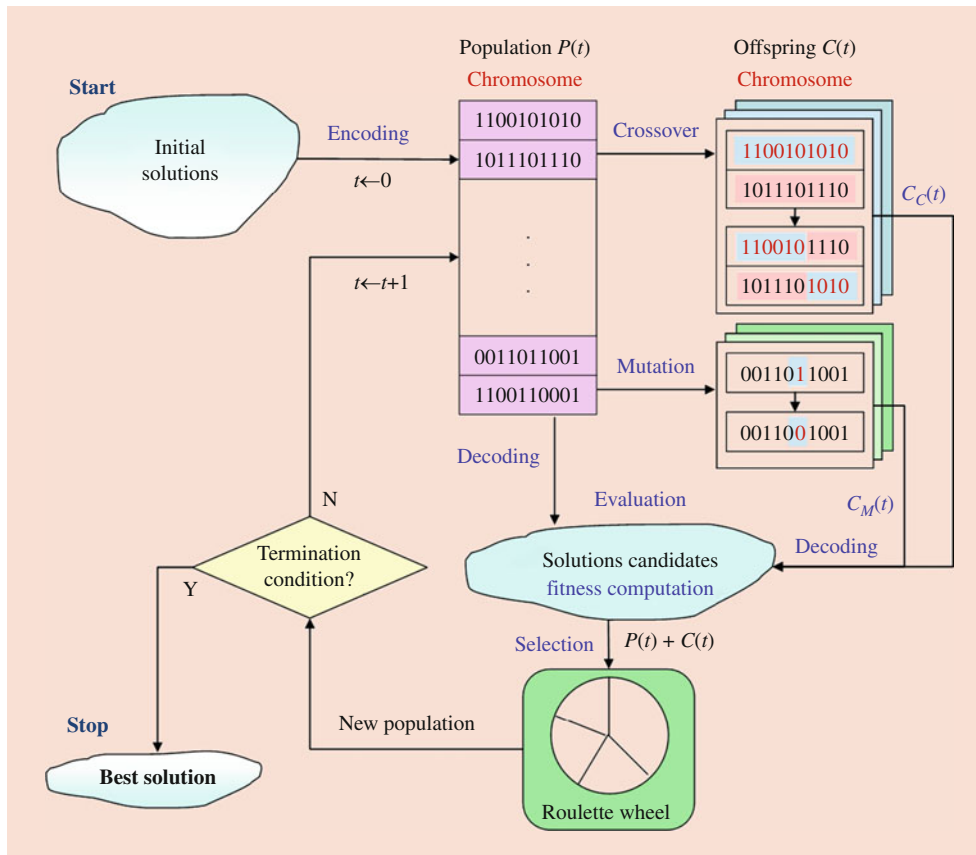


Fig. 33.1 The general structure of genetic algorithms

```

while (not terminating condition) do
  create  $C(t)$  from  $P(t)$  by crossover routine;
  create  $C(t)$  from  $P(t)$  by mutation routine;
  evaluate  $C(t)$  by decoding routine & update the best solution;
  reproduce  $P(t + 1)$  from  $P(t)$  and  $C(t)$  by selection routine;
   $t \leftarrow t + 1$ ;
end

```

output: the best solution;

end;

Crossover is the main genetic operator. It operates on two chromosomes at a time and generates offspring by combining both chromosomes' features. A simple way to achieve crossover would be to choose a random cut-point and generate the offspring by combining the segment of one parent to the left of the cut-point with the segment of the other parent to the right of the cut-point.

Mutation is a background operator, which produces spontaneous random changes in various chromosomes. A simple way to achieve mutation would be to alter one or more genes.

33.1.2 Hybrid Genetic Algorithms

Genetic algorithms (GAs) have proved to be a versatile and effective approach for solving optimization problems. Nevertheless, there are many situations in which the simple GA does not perform particularly well, and various methods have been proposed [5]. One of the most common forms of hybrid genetic algorithms is to incorporate local search (LS) or traditional optimization as an add-on extra to the canonical GA loop of recombination and selection [6–9]. With the hybrid approach, local optimization such as hill climbing is applied to each newly generated offspring to move it to a local optimum before injecting it into the population as shown in Fig. 33.2. Genetic algorithms are used to perform global exploration among the population, while heuristic methods are used to perform local exploitation around chromosomes. Because of the complementary properties of genetic algorithms and conventional heuristics, the hybrid approach often outperforms either method operating alone. Some work has been done to reveal the natural mechanism behind such a hybrid approach, among which is Lamarckian evolution. The hybrid GA can be combined with LS and tuned parameters

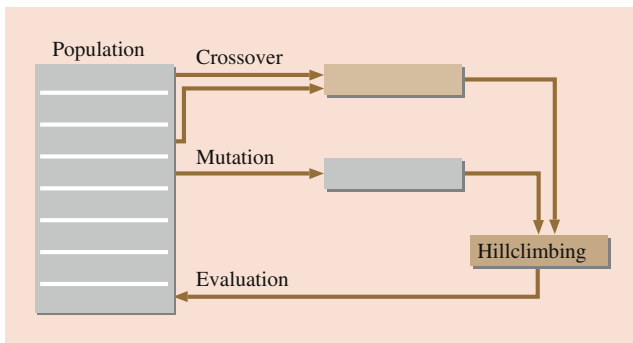


Fig. 33.2 General structure of hybrid genetic algorithms

by FLSI routines. The general structure of hybrid genetic algorithms is described as follows:

procedure: Hybrid Genetic Algorithm (HGA)

input: problem data, paramters

output: the best solution

begin

$t \leftarrow 0$

initialize $P(t)$ by encoding routine;

evaluate $P(t)$ by decoding routine & keep
the best solution;

while (not termination condition) **do**

create $C(t)$ from $P(t)$ by crossover routine;

create $C(t)$ from $P(t)$ by mutation routine;

evaluate $C(t)$ by decoding routine &
update best solution;

improve best solution by LS routine;

tune parameters by FLC routine;

reproduce $P(t + 1)$ from $P(t)$ and $C(t)$ by
selection routine;

$t \leftarrow t + 1$;

end

output: the best solution;

end;

33.1.3 Adaptive Genetic Algorithms

There are two basic approaches to applying the genetic algorithms to a given problem: (1) to adapt a problem to the genetic algorithms and (2) to adapt the genetic algorithms to a problem.

Genetic algorithms were first created as a kind of generic and weak method featuring binary encoding and binary genetic operators. This approach requires a modification of the original problem into an appropriate form suitable for the genetic algorithms, as shown in Fig. 33.3.

To overcome such problems, various nonstandard implementations of the genetic algorithm have been created for

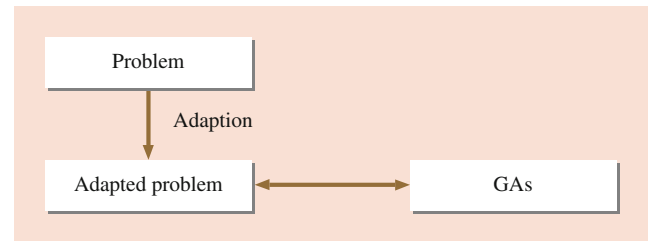


Fig. 33.3 Adapt a problem to the genetic algorithms

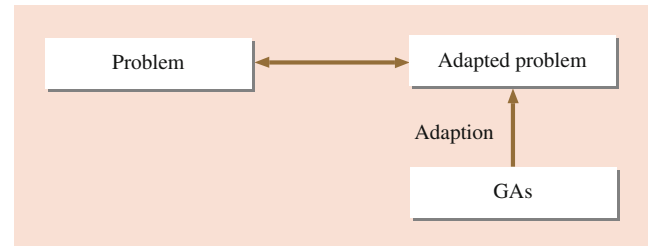


Fig. 33.4 Adapt the genetic algorithms to a problem

particular problems, which leave the problem unchanged and adapt the genetic algorithms by modifying a chromosome representation of a potential solution and applying appropriate genetic operators, as shown in Fig. 33.4. This approach has been successfully applied in the area of industrial engineering and is becoming the main approach in recent applications of genetic algorithms [4, 10, 11].

33.1.4 Fuzzy Logic Controller (FLC)

Fuzzy logic is much closer in spirit to human thinking and natural language than the traditional logical systems. In essence, the fuzzy logic controller (FLC) provides an algorithm which can convert a linguistic control strategy based on expert knowledge into an automatic control strategy. In particular, this methodology appears very useful when the processes are too complex for analysis by conventional techniques or when the available sources of information are interpreted qualitatively, inexactly, or with uncertainty [7–9].

Lin and Gen reported auto-tuning strategy by using fuzzy logic control for evolutionary algorithms adaptively regulation for taking the balance among the stochastic search and local search probabilities based on the change of the average fitness of parents and offspring in each generation [9].

The pioneering work to extend the fuzzy logic technique to adjust the strategy parameters of genetic algorithms dynamically was carried out by Yun, Lin, and Gen [7–9]. The experts' knowledge is stored in the knowledge base in the form of linguistic control rules. The inference system is the kernel of the controller, which provides an approximate reasoning based on the knowledge base. The generic structure of a fuzzy logic controller is shown in Fig. 33.5.

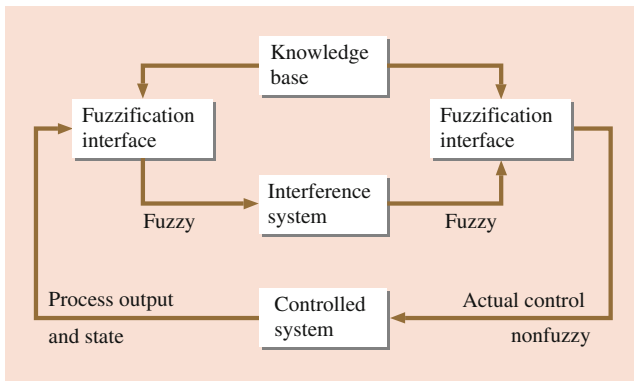


Fig. 33.5 A generic structure of a fuzzy logic controller

33.1.5 Multiobjective Optimization Problems

Many real-world decision-making problems involve multiple and conflicting objectives which need to be tackled while respecting various constraints, leading to overwhelming problem complexity. The multiobjective optimization problem (MOP) has been receiving growing interest from researchers with various backgrounds since early 1960. There are a number of scholars who have made significant contributions to the problem. Among them, Pareto is perhaps one of the most recognized pioneers in the MOP field [4, 12].

Multiobjective optimization problem with q objectives and m constraints will be formulated as follows:

$$\max [z_1 = f_1(\mathbf{x}), z_2 = f_2(\mathbf{x}), \dots, z_q = f_q(\mathbf{x})], \quad (33.1)$$

$$\text{s. t.} \quad g_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \dots, m. \quad (33.2)$$

$$\mathbf{x} \geq \mathbf{0} \quad (33.3)$$

where $\mathbf{x} \in \mathbf{R}^n$ is a vector of n decision variables, $f_j(\mathbf{x})$ is the objective function, and $g_i(\mathbf{x})$ are inequality constraint functions, which form the area of feasible solutions. We usually denote the feasible area in decision space with the set S as follows:

$$S = \{\mathbf{x} \in \mathbf{R}^n \mid g_i(\mathbf{x}) \leq 0, i = 1, \dots, m, \mathbf{x} \geq \mathbf{0}\} \quad (33.4)$$

We sometimes graph the multiple objective problem in both decision space and criterion space. S is used to denote the feasible region in the decision space, and Z is used to denote the feasible region in the criterion space as follows:

$$Z = \{\mathbf{z} \in \mathbf{R}^q \mid z_1 = f_1(\mathbf{x}), \dots, z_q = f_q(\mathbf{x}), \mathbf{x} \in S\} \quad (33.5)$$

where $\mathbf{z} \in \mathbf{R}^q$ is a vector of values of q objective functions. In other words, Z is the set of images of all points in S .

Although S is confined to the nonnegative orthant of \mathbf{R}^n , Z is not necessarily confined to the nonnegative orthant of \mathbf{R}^q .

Definition 1 For a given point $z_0 \in Z$, it is nondominated if and only if there does not exist another point $z \in Z$ such that, for the maximization case,

$$z_k > z_k^0, \text{ for some } k \in \{1, 2, \dots, q\} \text{ and} \quad (33.6)$$

$$z_l \geq z_l^0, \text{ for all } l \neq k \quad (33.7)$$

where z_0 is a dominated point in the criterion space Z .

A point in the decision space is efficient if and only if its image is a nondominated point in the criterion space Z .

Definition 2 For a given point $\mathbf{x}^0 \in S$, it is efficient if and only if there does not exist another point $\mathbf{x} \in S$ such that, for the maximization case,

$$f_k(\mathbf{x}) > f_k(\mathbf{x}^0), \text{ for some } k \in \{1, 2, \dots, q\} \text{ and} \quad (33.8)$$

$$f_l(\mathbf{x}) \geq f_l(\mathbf{x}^0), \text{ for all } l \neq k \quad (33.9)$$

where \mathbf{x}_0 is inefficient.

A point in the decision space is efficient if and only if its image is a nondominated point in the criterion space Z .

A. The Concept of a Pareto Solution

In most existing methods, Pareto solutions are identified at each generation and are only used to calculate fitness values or ranks for each chromosome. The overall structure of the basic multiobjective genetic algorithm (MoGA) is given as follows:

procedure: Multiobjective Genetic Algorithm

input: problem data, paramters

output: the best Pareto optimal solution

begin

$t \leftarrow 0$

initialize $P(t)$ by encoding routine;

calculate each $f_i(\mathbf{P})$ by decoding;

create Pareto frontier $E(\mathbf{P})$ by non-dominated;

evaluate $P(t)$ by fitness assignment function
& keep the best Pareto solution;

while (not termination condition) **do**
 create $C(t)$ from $P(t)$ by crossover routine;
 create $C(t)$ from $P(t)$ by mutation routine;
 calculate each $f_i(C)$ by decoding;
 update Pareto frontier $E(P)$ by non-dominated;
 evaluate $C(t)$ by fitness assignment function and update the best Pareto solution;
 reproduce $P(t+1)$ from $P(t)$ and $C(t)$ by selection routine;
 $t \leftarrow t+1$;
end
output: the best Pareto optimal solution;
end;

B. Fitness Assignment Function

Adaptive-Weight Genetic Algorithm (AWGA: Gen and Cheng 2000): Gen and Cheng proposed an adaptive-weight approach which utilizes some useful information from the current population to readjust weights to obtain a search pressure toward a positive ideal point.

For the examined solutions at each generation, we define two extreme points: the maximum extreme point z^+ and the minimum extreme point z^- in criteria space as follows:

$$z^+ = (z_1^{\max}, z_2^{\max}, \dots, z_q^{\max}), \quad (33.10)$$

$$z^- = (z_1^{\min}, z_2^{\min}, \dots, z_q^{\min}), \quad (33.11)$$

where z_k^{\max} and z_k^{\min} are the maximum value and minimum value for objective k in the current population. Let P denote the set of the current population. For a given individual x , the maximal value and minimal value for each objective are defined as follows:

$$z_k^{\max} = \max\{f_k(x) | x \in P\}, \quad k = 1, 2, \dots, q, \quad (33.12)$$

$$z_k^{\min} = \min\{f_k(x) | x \in P\}, \quad k = 1, 2, \dots, q, \quad (33.13)$$

The hyperparallelogram defined by the two extreme points is a minimal hyperparallelogram containing all current solutions. The two extreme points are renewed at each generation. The maximum extreme point will gradually approximate the positive ideal point. The adaptive weight for objective k is calculated by

$$w_k = \frac{1}{z_k^{\max} - z_k^{\min}}, \quad k = 1, 2, \dots, q. \quad (33.14)$$

For a given individual x , the weighted-sum objective function is given by

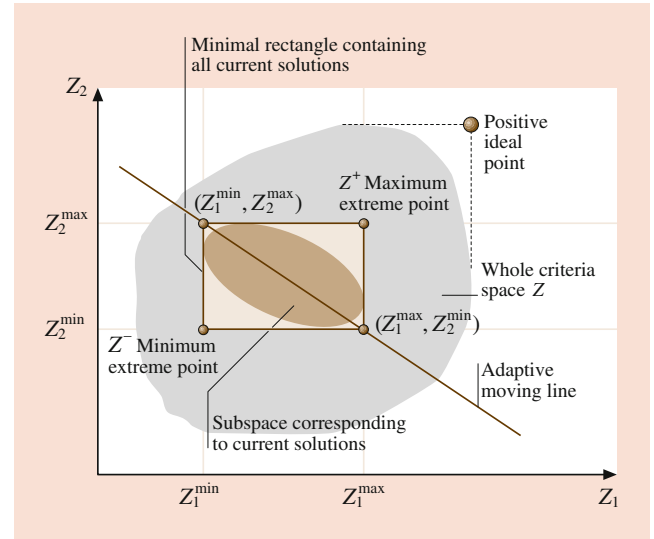


Fig. 33.6 Adaptive weights and adaptive hyperplane

$$eval(x) = \sum_{k=1}^q w_k (z_k - z_k^{\min}) \quad (33.15)$$

$$= \sum_{k=1}^q \frac{z_k - z_k^{\min}}{z_k^{\max} - z_k^{\min}} \quad (33.16)$$

$$= \sum_{k=1}^q \frac{f_k(x) - z_k^{\min}}{z_k^{\max} - z_k^{\min}}. \quad (33.17)$$

As the extreme points are renewed at each generation, the weights are renewed accordingly. Figure 33.6 is a hyperplane defined by the extreme points in the current solutions. It is an adaptive moving line defined by the extreme points (z_1^{\max}, z_2^{\min}) and (z_1^{\min}, z_2^{\max}) as shown Fig. 33.6. The rectangle defined by the extreme points (z_1^{\max}, z_2^{\min}) and (z_1^{\min}, z_2^{\max}) is the minimal rectangle containing all current solutions.

Nondominated Sorting Genetic Algorithm II (NSGA II: Deb 2001) Deb developed a Pareto ranking-based fitness assignment, and it is called NSGA [12]. In each method, the nondominated solutions constituting a nondominated front are assigned the same dummy fitness value. These solutions are shared with their dummy fitness values (phenotypic sharing on the decision vectors) and ignored in the further classification process. Finally, the dummy fitness is set to a value less than the smallest shared fitness value in the current non-dominated front. Then the next front is extracted. The procedure of NSGA II is repeated until all individuals in the population are classified [12].

Strength Pareto Evolutionary Algorithm 2 (SPEA2: Zitzler et al. [13]) Zitzler and Thiele proposed strength Pareto evolutionary algorithm and an extended version SPEA2 [13]

that combines several features of previous MoGA in a unique manner. The fitness assignment procedure is a two-stage process. The individuals in the external nondominated set P' are ranked.

Interactive Adaptive-Weight Genetic Algorithm (i-AWGA: Lin and Gen [9]) Lin and Gen proposed an interactive AWGA, which is an improved adaptive-weight fitness assignment approach with the consideration of the disadvantages of weighted-sum approach and Pareto ranking-based approach [14]. They combined a penalty term to the fitness value for all of dominated solutions. Firstly, we calculate the adaptive weight $w_i = 1/(z_i^{\max} - z_i^{\min})$ for each objective by using AWGA. Afterward, we calculate the penalty term $p(v_k) = 0$, if v_k is nondominated solution in the nondominated set P . Otherwise $p(v_k)=1$ for dominated solution v_k . Lastly, we calculate the fitness value of each chromosome by combining the i-AWGA method:

$$eval(v_k) = \sum_{i=1}^q w_i(z_i^k - z_i^{\min}) + p(v_k), \forall k \in \text{popSize} \quad (33.18)$$

Hybrid Sampling Strategy-Based EA Zhang et al. [15] Zhang et al. proposed a hybrid sampling strategy-based evolutionary algorithm [15, 16]. A Pareto dominating and dominated relationship-based fitness function (PDDR-FF) is proposed to evaluate the individuals. The PDDR-FF of an individual S_i is calculated by the following function:

$$eval(S_i) = q(s_i) + \frac{1}{p(s_i + 1)}, i = 1, 2, \dots, \text{popSize} \quad (33.19)$$

where $p()$ is the number of individuals which can be dominated by the individual S . $q()$ is the number of individuals which can dominate the individual S . The PDDR-FF can set the obvious difference values between the nondominated and dominated individuals. The general structure in the pseudocode of multiobjective hybrid genetic algorithm (Mo-HGA) is designed MoGA with a local search and FLC in HGA.

33.1.6 Genetic Algorithm with Learning

Earlier attention of the interaction between machine learning (ML) and genetic algorithm (GA) was presented by Goldberg and Holland in 1988 [1, 17]. During the search process, the GA can store ample data about the search information, population information, and problem features. Thus, the ML

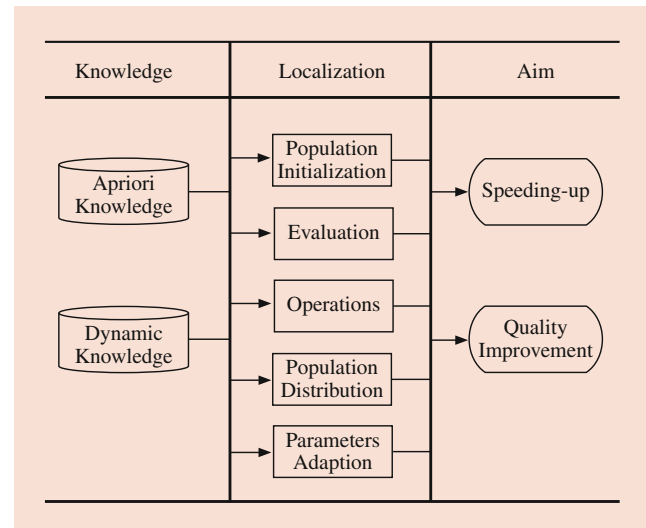


Fig. 33.7 An illustration of the hybridization taxonomy [18]

technique is helpful in analyzing these data for enhancing the search performance of GAs. Recently Lin and Gen surveyed hybrid evolutionary optimization algorithms with learning for various production scheduling problems [18]. In this way, useful information can be extracted to understand the search behavior and to assist with future searches for the global optimum. In many applications, GAs incorporating ML techniques have been proven to be advantageous in both convergence speed and solution quality. Jourdan et al. [19] and Zhang et al. [20] presented surveys of ML technique enhanced GAs.

The ML techniques can be incorporated into different GAs in various ways, and they affect GA also on various aspects. To classify the different ways to hybridization of ML techniques and GAs, Jourdan et al. presented a taxonomy by knowledge type/localization/aim as shown in Fig. 33.7.

Knowledge Type

ML as a knowledge discovery technique is the process of automatically exploring large volumes of data. For enhancing the performance of EAs, two kinds of knowledge can be distinguished:

- (1) A priori knowledge, a previously acquired knowledge, such as problem features and problem data set.
- (2) Dynamic knowledge, a dynamically acquired knowledge which is extracted or discovered during the search, such as search information and population information.

Aim

Another useful information to classify algorithms is to distinguish the aim of the hybridization:

- (1) Speeding-up: The hybridization is used to reduce the computational time, i.e., speed up techniques, by simplification of the fitness, i.e., fitness approximation, or by significantly reducing the search space, e.g., leading the EAs in promising area
- (2) Quality improvement: The hybridization is used to improve the quality of the search by introducing knowledge in EA operations or in other parts of the EAs.

In fact, the insertion of ML techniques often leads to both speeding up the EAs and improving the quality.

Localization

Different EAs have similar framework in implementation and algorithmic characteristics. Michalewicz [5] summarized five basic components of EAs, and a general evolution process of the EAs is shown in Fig. 33.8:

- An evolutionary representation of potential solutions to the problem.
- A way to create a population (an initial set of potential solutions).
- An evaluation function rating solutions in terms of their fitness.
- Evolutionary operators that alter the genetic composition of offspring (crossover, mutation, selection, etc.).
- Parameter values that evolutionary algorithm uses (population size, probabilities of applying evolutionary operators, etc.).

33.2 Combinatorial Optimization Problems

Combinatorial optimization studies problems which are characterized by a finite number of feasible solutions. An impor-

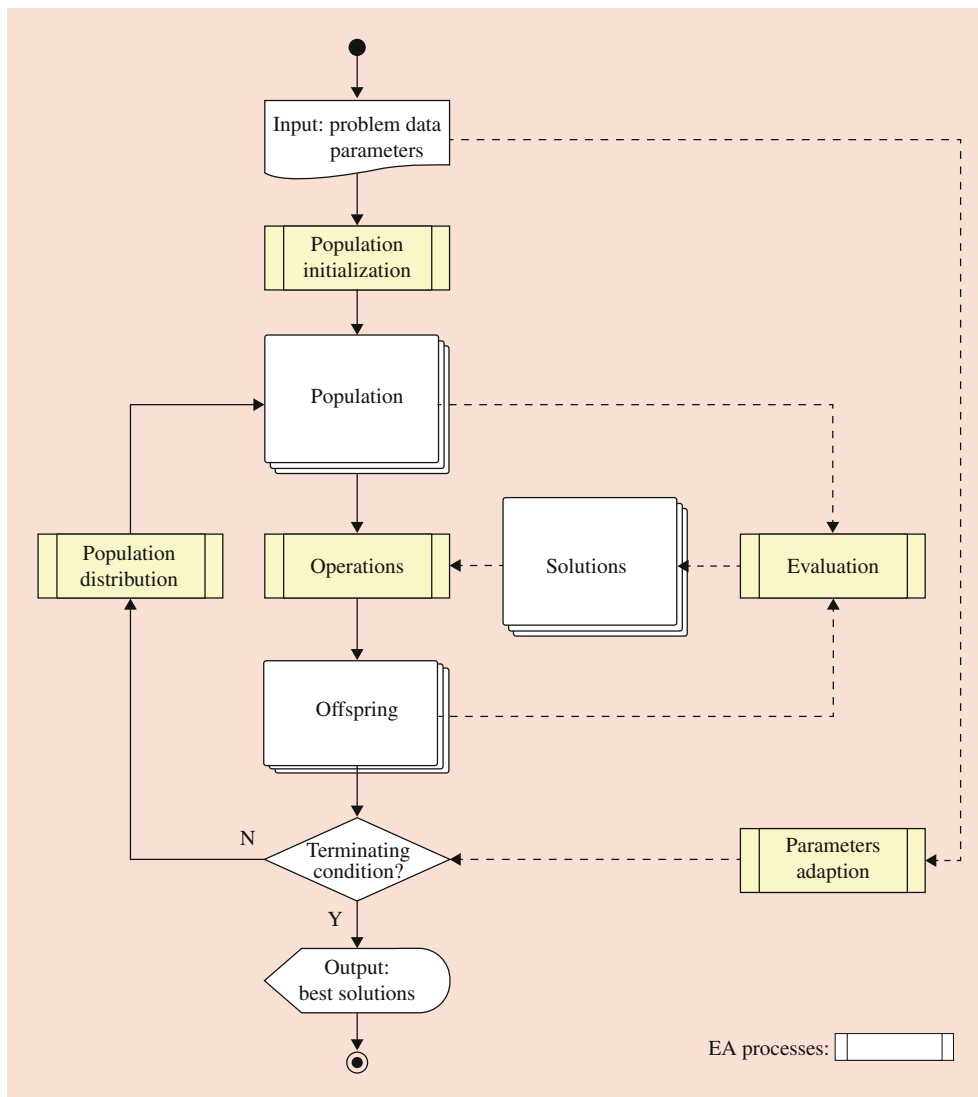


Fig. 33.8 A general evolution process of the EAs [18]

tant and widespread area of application concerns the efficient use of scarce resources to increase productivity.

33.2.1 Knapsack Problem

Suppose that we want to fill up a knapsack by selecting some objects among various objects (generally called items). There are n different items available, and each item j has a weight of w_j and amazing profit of p_j . The knapsack can hold a weight of at most W . The problem is to find an optimal subset of items so as to maximize the total profit subject to the knapsack's weight capacity. The profits, weights, and capacity are positive integers [21].

Let x_j be binary variables given by

$$x_j = \begin{cases} 1 & \text{if item } j \text{ is selected,} \\ 0 & \text{otherwise.} \end{cases} \quad (33.20)$$

The knapsack problem can be mathematically formulated as

$$\max \sum_{j=1}^n p_j x_j, \quad (33.21)$$

$$\text{s. t. } \sum_{j=1}^n w_j x_j \leq W, \quad (33.22)$$

$$x_j = 0 \text{ or } 1 \quad j = 1, 2, \dots, n. \quad (33.23)$$

Binary Representation Approach

The binary string is a natural representation for the knapsack problem, where one means the inclusion and zero the exclusion of one of the n items from the knapsack. For example, a solution for the 10-item problem can be represented as the following bit string:

$$\mathbf{x} = (x_1 x_2 \dots x_{10}) \\ (0 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0),$$

meaning that items 2, 4, and 9 are selected for inclusion in the knapsack.

33.2.2 Minimum Spanning Tree Model

Consider a connected undirected graph $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$ is a finite set of vertices representing terminals or telecommunication stations, etc. and $E = \{e_{ij} | e_{ij} = (v_i, v_j), v_i, v_j \in V\}$ is a finite set of edges representing connections between these terminals or stations. Each edge has an associated positive real number denoted by $W = \{w_{ij} | w_{ij} = w(v_i, v_j), w_{ij} > 0, v_i, v_j \in V\}$ representing distance, cost, and

so on. The vertices and edges are sometimes referred to as nodes and links, respectively.

Based on their different backgrounds, many researchers have proposed varieties of spanning tree problems with some constraints on them, such as the spanning tree problem with a degree constraint, the stochastic spanning tree problem, the quadratic spanning tree problem, the multicriteria spanning tree problem, and the spanning tree problem with a constraint on the number of leaves or leaf-constrained spanning tree problem [22, 23].

A spanning tree is a minimal set of edges from E that connects all the vertices in V , and therefore at least one spanning tree can be found in graph G . The minimum spanning tree is just one of the spanning trees whose total weight of all edges is minimal. It can be formulated as

$$\min z(\mathbf{x}) = \sum_{i=1}^{n-1} \sum_{j=2}^n w_{ij} x_{ij}, \quad (33.24)$$

$$\text{s. t. } \sum_{i=1}^{n-1} \sum_{j=2}^n x_{ij} = n - 1; \quad (33.25)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1, \quad S \subseteq V \setminus \{1\}, \quad |S| \geq 2, \quad (33.26)$$

$$x_{ij} = 0 \text{ or } 1, \quad i = 1, 2, \dots, n-1, \\ j = 2, 3, \dots, n, \quad (33.27)$$

where

$$x_{ij} = \begin{cases} 1, & \text{if edge } (i, j) \text{ is selected in a spanning tree} \\ 0, & \text{otherwise} \end{cases} \quad (33.28)$$

and T is a set of the spanning trees of graph G .

A. Tree Encodings

For the minimum spanning tree (MST) problem, the method of encoding a tree is critical for the genetic algorithm approach because the solution should be a tree.

If we associate an index k with each edge, i.e., $E = \{e_k\}$, $k = 1, 2, \dots, K$, where k is the number of edges in a graph, a bit string can represent a candidate solution by indicating which edges are used in a spanning tree, as illustrated in Fig. 33.9.

B. Genetic Algorithm Approach

Representation The chromosome representation for a spanning tree should contain, implicitly or explicitly, the degree on each vertex. Among the several tree encodings, only the Prüfer number encoding explicitly contains the information of vertex degree, i.e., that any vertex with degree d will appear

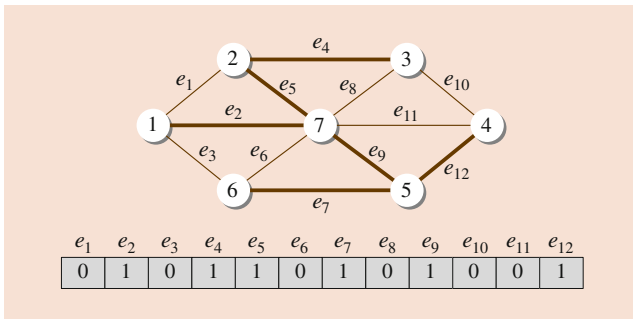


Fig. 33.9 A graph with its edge encoding for a spanning tree

exactly $d-1$ times in the encoding. Thus the Prüfer number encoding is adopted.

Crossover and Mutation Prüfer number encoding can still represent a tree after any crossover or mutation operations. Simply, the one-point crossover operator is used, as illustrated in Fig. 33.10. Mutation is performed as random perturbation within the permissive integer from 1 to n (n is the number of vertices in graph). An example is given in Fig. 33.11.

33.2.3 Set-Covering Model

The problem is to cover the rows of an m -row/ n -column zero-one matrix by a subset of columns at minimal cost. Considering a vector n , x_j is 0–1 variable that takes on the 3 value 1, if item j is selected (with a cost $c_j > 0$). The set-covering problem is then formulated as

$$\min z(\mathbf{x}) = \sum_{j=1}^n c_j x_j, \tag{33.29}$$

$$\text{s. t. } \sum_{j=1}^n a_{ij} x_j \leq 1 \quad i = 1, 2, \dots, m, \tag{33.30}$$

$$x_j \in \{0, 1\}, \quad j = 1, 2, \dots, n. \tag{33.31}$$

Genetic Algorithm Approach

Representation The fitness $f(\mathbf{x})$ of an individual is calculated simply by

$$f(\mathbf{x}) = \sum_{j=1}^n c_j x_j. \tag{33.32}$$

The initial population can be generated randomly.

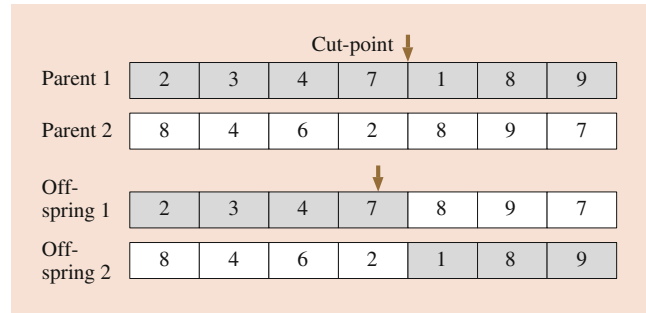


Fig. 33.10 Illustration of the crossover operation

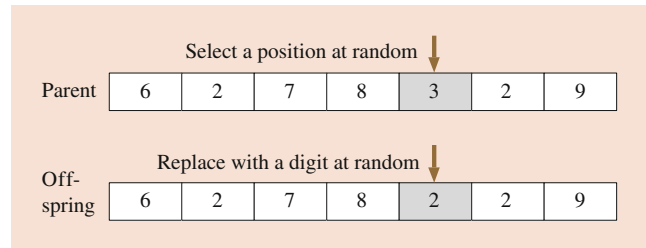


Fig. 33.11 Illustration of the mutation operation

Genetic Operators Beasley and Chu proposed a generalized fitness-based crossover operator called the fusion operator.

Let P_1 and P_2 be the parent strings. Let f_{P_1} and f_{P_2} be the fitness values of the parent strings P_1 and P_2 , respectively. Let C be the child string. The fusion operator works as follows:

Procedure: Fusion Operator

- Step 1. $i = 1$.
- Step 2. If $P_1[i] = P_2[i]$, then $C[i] \leftarrow P_1[i] = P_2[i]$.
- Step 3. If $P_1[i] \neq P_2[i]$, then
 - (a) $C[i] \leftarrow P_1[i]$ with probability $p = f_{P_2} / (f_{P_1} + f_{P_2})$.
 - (b) $C[i] \leftarrow P_2[i]$ with probability $1-p$
- Step 4. If $i = n$, stop; otherwise, set $i \leftarrow i + 1$ and go to step 1.

33.2.4 Bin-Packing Model

The bin-packing problem consists of placing n objects into a number of bins (at most n bins). Each object has a weight ($w_i > 0$) and each bin has a limited bin capacity ($c_i > 0$). The problem is to find the best assignment of objects to bins such that the total weight of the objects in each bin does not exceed its capacity and the number of bins used is minimized.

Genetic Algorithm Approach

Representation The most straightforward approach is to encode the membership of objects in the solution. For instance,

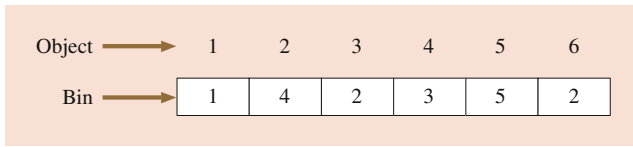


Fig. 33.12 Representation of membership of objects

the chromosome [1 4 2 3 5 2] would encode a solution where the first object is in bin 1, the second in bin 4, the third in bin 2, the fourth in bin 3, the fifth in bin 5, and the sixth in bin 2. This representation for the bin-packing problem is illustrated in Fig. 33.12.

A mathematical formulation for the bin-packing problem is given as follows [10, 21]:

$$\min z(y) = \sum_{i=1}^n y_i, \tag{33.33}$$

$$\text{s. t. } \sum_{j=1}^n w_j x_{ij} \leq c_i y_i, \quad i \in N = \{1, 2, \dots, n\}, \tag{33.34}$$

$$\sum_{i=1}^n x_{ij} = 1, \quad j \in N \tag{33.35}$$

$$y_i = 0 \text{ or } 1, \quad i \in N \tag{33.36}$$

$$x_{ij} = 0 \text{ or } 1, \quad i, j \in N \tag{33.37}$$

where

$$y_i = \begin{cases} 1, & \text{if bin } i \text{ is used} \\ 0, & \text{otherwise,} \end{cases} \tag{33.38}$$

$$x_{ij} = \begin{cases} 1, & \text{if object } j \text{ is assigned to bin } i \\ 0, & \text{otherwise.} \end{cases} \tag{33.39}$$

Genetic Operators

Procedure: Crossover

- Step 1 Select at random two crossing sites, delimiting the crossing section, in each of the two parents.
- Step 2 Inject the contents of the crossing section of the first parent at the first crossing site of the second parent.
- Step 3 Eliminate all objects now occurring twice from the bins they were members of in the second parent, so that the *old* membership of these objects gives way to the membership specified by the *new* injected bins. Consequently, some of the old groups coming from the second parent are altered.
- Step 4 If necessary, adapt the resulting bins, according to the hard constraints and the cost function to optimize.

Step 5 Apply steps 2–4 to the two parents with their roles permuted to generate the second child.

33.2.5 Traveling-Salesman Model

The traveling-salesman problem (TSP) is one of the most widely studied combinatorial optimization problems. Its statement is deceptively simple: a salesman seeks the shortest tour through *n* cities.

For example, a tour of a nine-city TSP

$$3 - 2 - 5 - 4 - 7 - 1 - 6 - 9 - 8$$

is simply represented as follows:

$$[3 \ 2 \ 5 \ 4 \ 7 \ 1 \ 6 \ 9 \ 8].$$

This representation is also called a path representation or order representation. This representation may lead to illegal tours if the traditional one-point crossover operator is used; therefore many crossover operators have been investigated for it. Another method is the random keys representation. This representation encodes a solution with random numbers from (0,1). These values are used as sort keys to decode the solution.

For example, a chromosome for a nine-city problem may be

$$[0.23 \ 0.82 \ 0.45 \ 0.74 \ 0.87 \ 0.11 \ 0.56 \ 0.69 \ 0.78]$$

where position *i* in the list represents city *i*. The random number in position *i* determines the visiting order of city *i* in a TSP tour. We sort the random keys in ascending order to get the following tour:

$$6 - 1 - 3 - 7 - 8 - 4 - 9 - 2 - 5.$$

Genetic Algorithm Approach *Representation*: Permutation representation is perhaps the most natural representation of a TSP tour, where cities are listed in the order in which they are visited [24, 25].

Crossover Operators

Procedure Partial-Mapped Crossover (PMX) [25].

- Step 1. Select two positions along the string uniformly at random.
- Step 2. Exchange two substrings between parents to produce proto-children.
- Step 3. Determine the mapping relationship between two mapping sections.
- Step 4. Legalize offspring with the mapping relationship.

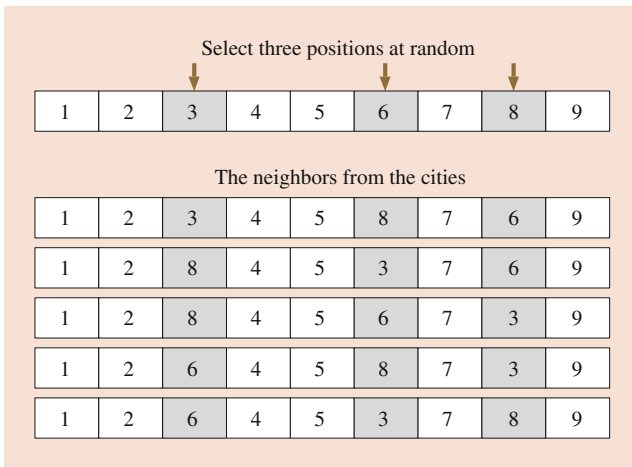


Fig. 33.13 Illustration of the heuristic mutation operator

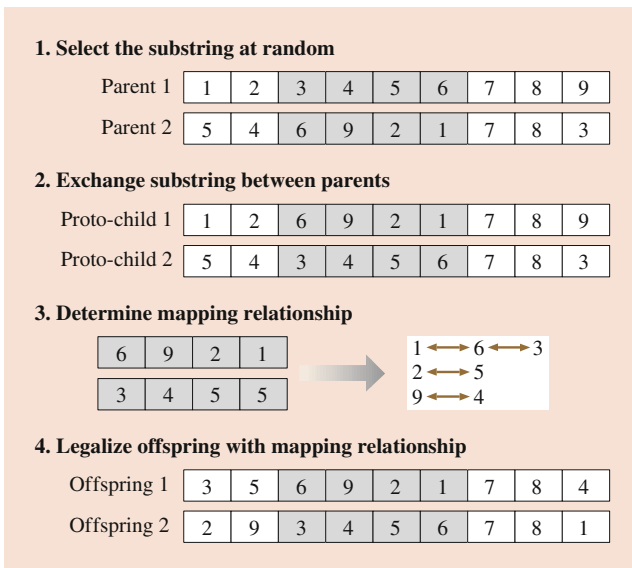


Fig. 33.14 Illustration of the PMX operator

The procedure is illustrated in Fig. 33.13.

Mutation Operators

Procedure : *Heuristic Mutation* [26,27].

- Step 1. Pick n genes at random.
- Step 2. Generate neighbors according to all possible permutation of the selected genes.
- Step 3. Evaluate all neighbors and select the best one as offspring.

The procedure is illustrated in Fig. 33.14.

33.3 Network Design Problems

Network design and routing are one of important issues in the building and expansion of computer networks. Many ideas

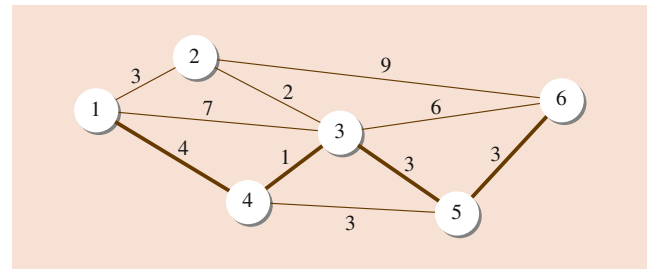


Fig. 33.15 Simple undirected graph with six nodes and ten edges

and methods have been proposed and tested in the past two decades. Recently, there is an increasing interest in applying genetic algorithms to problems related to computer network [4,28].

33.3.1 Shortest-Path Model

An undirected graph $G = (V, E)$ comprises a set of nodes $V = \{1, 2, \dots, n\}$ and a set of edges $E \in V \times V$ connecting nodes in V .

Corresponding to each edge, there are two nonnegative numbers c_{ij}^1 and c_{ij}^2 representing the cost and distance, or other items of interest, from node i to node j is a sequence of edges $(i, l), (l, m), \dots, (k, j)$ from E in which no node appears more than once. A path can also be equivalently represented as a sequence of nodes (i, l, m, \dots, k, j) . For the example given in Fig. 33.15, $(1,4), (4,3), (3,5), (5,6)$ is a path from node 1 to node 6. The node representation is $(1, 4, 3, 5, 6)$.

Let 1 denote the initial node and n denote the end node of the path. Let x_{ij} be an indicator variable defined as follows:

$$x_{ij} = \begin{cases} 1, & \text{if edge } (i, j) \text{ is included in the path} \\ 0, & \text{otherwise.} \end{cases} \tag{33.40}$$

Genetic Algorithm Approach

Priority-Based Encoding [14,29–32] The position of a gene is used to represent a node, and the value is used to represent the priority of the node for constructing a path among the candidates. The encoding method is denoted by priority-based encoding. The path corresponding to a given chromosome is generated by a sequential node appending procedure, beginning from the specified node 1 and terminating at the specified node n . The bicriteria shortest-path problem can be formulated as follows:

$$\min z_1(\mathbf{x}) = \sum_i \sum_j c_{ij}^1 x_{ij}, \tag{33.41}$$

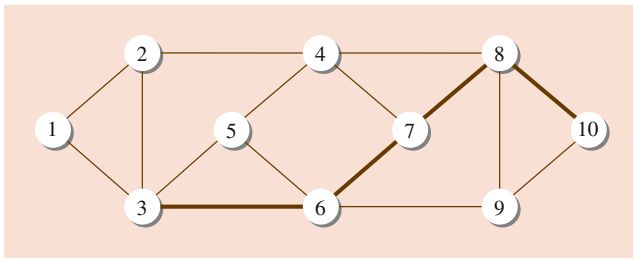


Fig. 33.16 Simple undirected graph with 10 nodes and 16 edges

Position: node ID	1	2	3	4	5	6	7	8	9	10
Value: priority	7	3	4	6	2	5	8	10	1	9

Fig. 33.17 Example of priority-based encoding

$$\min z_2(\mathbf{x}) = \sum_i \sum_j c_{ij}^2 x_{ij}, \quad (33.42)$$

$$\text{s. t. } \sum_j x_{ij} \leq 2, \forall i \in V, \quad (33.43)$$

$$\sum_j x_{ij} \geq x_{ik}, \forall (i, k) \in E, \forall i \in V \setminus \{1, n\} \quad (33.44)$$

$$\sum_j x_{1j} = \sum_j x_{jn} = 1, \forall i, j \in V, \quad (33.45)$$

$$x_{ij} = x_{ji}, \quad \forall (i, j) \in E, \quad (33.46)$$

$$0 \leq x_{ij} \leq 1, \quad \forall (i, j) \in E \quad (33.47)$$

Consider the undirected graph shown in Fig. 33.16. Suppose we are going to find a path from node 1 to node 10. An encoding of the instance is given in Fig. 33.17. At the beginning, we try to find a node for the position next to node 1. Nodes 2 and 3 are eligible for the position, which can be easily fixed according to the adjacency relation among nodes. The priorities of them are 3 and 4, respectively. Node 3 has the highest priority and is put into the path. The possible nodes next to node 3 are nodes 2, 5, and 6. Because node 6 has the largest priority value, it is put into the path. Then we form the set of nodes available for the next position and select the one with the highest priority among them. These steps are repeated until we obtain a complete path (1, 3, 6, 7, 8, 10).

For an n -node problem, let Ω be a set containing integer exclusively from the set Ω , that is, $\Omega = \{1, 2, \dots, n\}$; let p_i denote the priority for node i , which is a random integer exclusively from the set Ω . Priorities p_i of all nodes satisfy the following conditions:

$$p_i \neq p_j, \quad p_i, p_j \in \Omega, \quad i \neq j, i, j \neq 1, 2, \dots, n \quad (33.48)$$

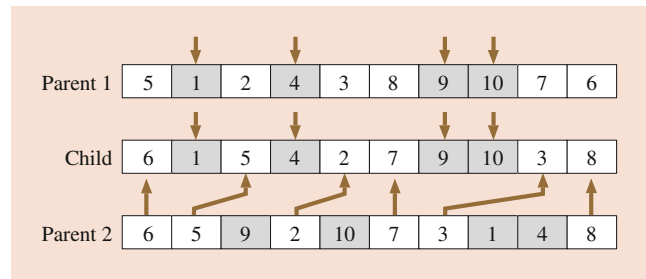


Fig. 33.18 Position-based crossover operator

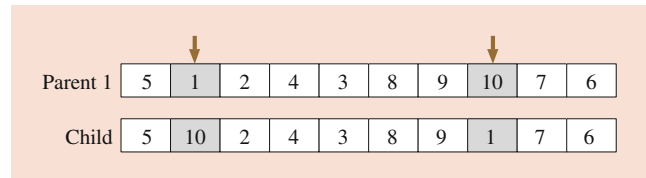


Fig. 33.19 Swap mutation operator

Then the priority-based encoding can be formally defined as

$$[p_1 \ p_2 \ \dots \ p_n]$$

Genetic Operators

Here the position-based crossover operator proposed by Syswerda is adopted. It can be viewed as a kind of uniform crossover operator for integer permutation representation together with a pairing procedure, as shown in Fig. 33.18. Essentially, it takes some genes from one parent at random and fills the vacuum position with genes from the other parent using a left-to-right scan. The swap mutation operator is used here, which simply selects two positions at random and swaps their contents as shown in Fig. 33.19.

33.3.2 Maximum-Flow Model

There have been many applications of the maximum-flow problem (MXF) in the real world. One of them is to determine the maximum flow through a pipeline network. Assume that oil should be shipped from the refinery (the source) to a storage facility (the sink) along arcs of the network. Each arc has a capacity which limits the amount of flow along that arc. Here, we want to determine the largest possible.

This is a flow that can be sent from the refinery to the storage facility with the restriction that no arc (pipe) capacity can be exceeded. MXF has also been applied to some other applications such as the problem of selecting sites for an electronic message-transmission system and dynamic flows in material-handling systems [14, 32, 33]. A mathematical formulation for the bin-packing problem is given by

$$\max f \quad (33.49)$$

$$\text{s. t. } \sum_{j=1}^n x_{ij} - \sum_{k=1}^m x_{ki} = \begin{cases} f, & \text{if } i = 1 \\ 0, & \text{if } i = 2, 3, \dots, m-1 \\ -f, & \text{if } i = m \end{cases} \quad (33.50)$$

$$0 \leq x_{ij} \leq u_{ij}, \quad i, j = 1, 2, \dots, m \quad (33.51)$$

where f is the amount of flow in the network from node 1 to node m and u_{ij} is arc capacities.

Genetic Algorithm Approach

The priority-based encoding method is used to represent the chromosome. The chromosome here is represented by m -digit numbers that are generated randomly. Each number represents the priority of the node.

Crossover As the first step in the crossover operation, we generate random numbers γ_k in the range $[0, 1]$ ($k = 1, 2, \dots, popSize$). Next, we select the chromosomes v_k to which the crossover operation will be applied. If $\gamma_k < p_C$, then the crossover operation will be applied to chromosome v_k .

Mutation Similarly, the first step in the mutation operation is to generate a random γ_r in the range $[0, 1]$, ($r = 1, 2, \dots, popSize$).

If $\gamma_r < p_M$, then the chromosome v_k ($l = (r/m + 1)$) is chosen for the mutation operation.

33.3.3 Minimum-Cost-Flow Model

The minimum-cost-flow (MCF) problem is known as a useful type of network optimization problem. It consists of finding the minimum-cost flows in the networks. For this problem, we are given a directed network $G = (X, A)$ in which each arc connecting nodes i and j in the network is associated with a cost c_{ij} and a capacity u_{ij} . A feasible solution to the MCF problem should satisfy two constraints. First, the flow through each arc should satisfy the capacity constraint. Second, the conservation of flow in all nodes should also be preserved. The conservation of flows here means that the flow into a node must equal the flow out of the node. The common objective is to determine the feasible network flow that minimizes the total cost.

A mathematical formulation for the bin-packing problem is given by

$$\min z = \sum_{i=1}^m \sum_{j=1}^m c_{ij} x_{ij}, \quad (33.52)$$

$$\text{s. t. } \sum_{j=1}^m x_{ij} - \sum_{k=1}^m x_{ki} = b_i, \quad i = 1, \dots, m, \quad (33.53)$$

$$x_{ij} \geq 0, \quad i, j = 1, \dots, m, \quad (33.54)$$

where x_{ij} is the flow through an arc and c_{ij} is the unit shipping cost along the arc. Equation (33.53) is called the flow conservation or Kirchhoff equation and indicates that flow may be neither created nor destroyed in the network. Genetic Algorithm Approach

Representation The chromosome here is represented by m -digit numbers generated randomly. Each number represents the priority of the node respectively.

Crossover The crossover is done by selecting two chromosome randomly. We use the partially matched crossover (PMX) method for the crossover operation.

Mutation Mutation here is done by selecting a chromosome at random. Two bit positions of the chromosome are exchanged.

33.3.4 Centralized Network Design

Consider a complete, undirected graph $G = (V, E)$; let $V = \{1, 2, \dots, n\}$ be the set of nodes representing terminals. Denote the central site or root node as node 1, and let $E = \{(i, j) | i, j \in V\}$ be the set of edges representing all possible telecommunication wiring. For a subset of nodes $S (\subseteq V)$, define $E(S) = \{(i, j) | i, j \in S\}$ as the edges whose endpoints are both in S . Define the following binary decision variables for all edges $(i, j) \in E$:

$$x_{ij} = \begin{cases} 1, & \text{if edge } (i, j) \text{ is selected} \\ 0, & \text{otherwise.} \end{cases} \quad (33.55)$$

Let c_{ij} be the fixed cost with respect to edge (i, j) in the solution, and suppose that d_i represents the demand at each node $i \in V$, where by convention the demand of the root node is $d_1 = 0$. Let $d(S)$, $S \subseteq V$ denote the sum of the demands of nodes of S . The subtree capacity is denoted with k . The centralized network design problem can be formulated as follows:

$$\min z = \sum_{i=1}^{n-1} \sum_{j=2}^n c_{ij} x_{ij}, \quad (33.56)$$

$$s. t. \sum_{i=1}^{n-1} \sum_{j=2}^n x_{ij} = 2(n - 1), \quad (33.57)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq 2[|S| - \lambda(S)]$$

$$S \subseteq V \setminus \{1\}, |S| \geq 2, \quad (33.58)$$

$$\sum_{i \in U} \sum_{j \in U} x_{ij} \leq 2(|U| - 1), U \subset V,$$

$$|U| \geq 2, \{1\} \in U, \quad (33.59)$$

$$x_{ij} = 0 \text{ or } 1, i = 1, 2, \dots, n - 1, \\ j = 2, 3, \dots, n, \quad (33.60)$$

Equality (33.57) is true of all spanning trees: a tree with n nodes must have $n - 1$ edges. Inequality (33.58) is a standard inequality for spanning trees: if more than $|U| - 1$ edges connect the nodes of a subset U , then the set U must contain a cycle. The parameter $\lambda(S)$ refers to the bin-packing number of set S , namely, the number of bins of size k needed to pack the nodes of items of size d_i for all $i \in S$. These constraints are similar to those for inequality (33.59), except that they reflect the capacity constraint: if the set S does not contain the root node, then the nodes of S must be contained in at least $\lambda(S)$ different subtrees of the root.

In Fig. 33.20, node ID is the node number based on the depth-first search (DFS), and the degree at node ID is the number of connecting nodes.

Genetic Algorithm Approach

To solve the centralized network design problem by using a genetic algorithm, a tree-based permutation encoding method is adopted to encode the candidate solutions, as illustrated in Fig. 33.20.

33.3.5 Multistage Process Planning Model

The multistage process planning (MPP) system usually consists of a series of machining operations, such as turning, drilling, grinding, finishing, and so on, to transform a part into its final shape or product. The whole process can be divided into several stages. At each stage, there are a set of similar manufacturing operations. The MPP problem is to find the optimal process planning among all possible alternatives given certain criteria such as minimum cost, minimum time, and maximum quality or under several of these criteria.

For an n -stage MPP problem, let s_k be some state at stage k , $D_k(s_k)$ be the set of possible states to be chosen at stage k , and $k = 1, 2, \dots, n$, x_k be the decision variable to determine which state to choose at stage k ; obviously $x_k \in D_k(s_k)$, $k = 1, 2, \dots, n$. Then the MPP problem can be formulated as follows:

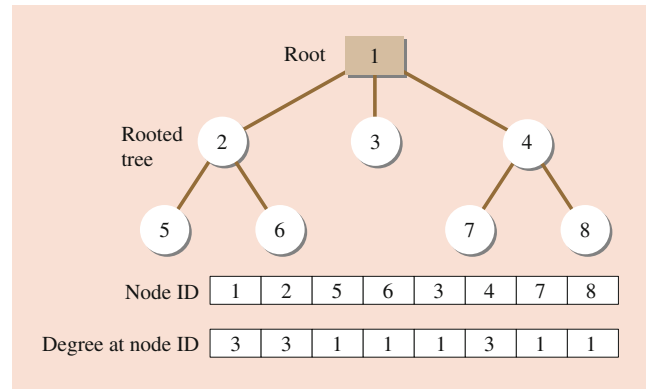


Fig. 33.20 Rooted tree and its tree-based permutation



Fig. 33.21 Mutation with neighborhood search

$$\min_{x_k \in D_k(s_k), k=1,2,\dots,n} V(x_1, x_2, \dots, x_n) = \sum_{k=1}^n v_k(s_k, x_k). \quad (33.61)$$

where $v_k(s_k, x_k)$ represents the criterion to determine x_k under state s_k at stage k , usually defined as a real number such as cost, time, or distance.

Genetic Algorithm Approach

Representation The MPP solution can be concisely encoded in a state permutation format by concatenating all the set states of stages. This state permutation encoding has a one-to-one mapping for the MPP problem. The probability of randomly producing a process planning is definitely 1. It is also easy to decode and evaluate. As to the initial population for an n -stage MPP problem, each individual is a permutation with $n - 1$ integers, whereas the integers are generated randomly with the number of all possible states in the corresponding stage.

Genetic Operation: In Zhou and Gen’s method, only the mutation operation was adopted because it is easy to hybrid the neighborhood search technique to produce more adapted offspring[10]. This hybrid mutation operation provides a great chance to evolve to the optimal solution.

Figure 33.21 shows an example for this mutation operation with a neighborhood search technique supposing that

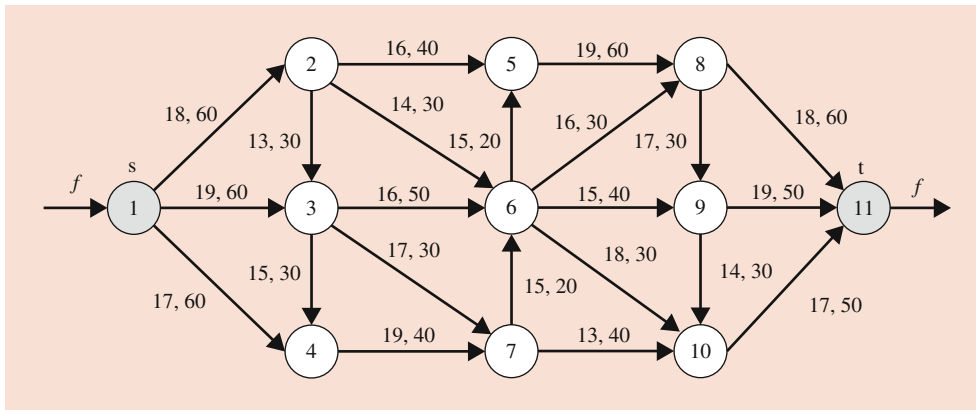


Fig. 33.22 A simple network with 11 nodes and 22 edges

the gene is at stage 3 and the number of possible states to be chosen is 4.

33.3.6 Bicriteria MXF/MCF Model

The bicriteria network design (BND) problem is known as NP-hard; it is not simply an extension from single objective to two objectives. Considering the characteristic of priority-based encoding method, we applied a new crossover operator called as weight mapping crossover (WMX), and insertion mutation operator is adopted [34].

Mathematical Formulation

Consider a directed network $G = (N, A)$, consisting of a finite set of nodes $N = \{1, 2, \dots, n\}$ and a set of directed arcs $A = \{(i, j), (k, l), \dots, (s, t)\}$ joining pairs of nodes in N . Arc (i, j) is said to be incident with nodes i and j and is directed from node i to node j . We shall assume that the network has n nodes and m arcs. Figure 33.22 presents a simple network with 11 nodes and 22 arcs.

The decision variables in the BND problem are the maximum possible flow z_1 with minimum cost z_2 from source node 1 to sink node n . Mathematically, this problem is formulated as follows (where summations are taken over existing arcs):

$$\max z_1 = f \tag{33.62}$$

$$\min z_2 = \sum_{i=1}^n \sum_{j=1}^n c_{ij}x_{ij} \tag{33.63}$$

$$\text{s. t. } \sum_{j=1}^n x_{ij} - \sum_{k=1}^n x_{ki} = \begin{cases} f & (i = 1) \\ 0 & (i = 2, 3, \dots, n - 1) \\ -f & (i = n) \end{cases} \tag{33.64}$$

Locus :	1	2	3	4	5	6	7	8	9	10	11
Node ID :	11	1	10	3	8	9	5	7	4	2	6

Fig. 33.23 An example of priority-based chromosome

$$0 \leq x_{ij} \leq u_{ij} \quad \forall (i, j) \tag{33.65}$$

$$x_{ij} = 0 \text{ or } 1 \quad \forall i, j \tag{33.66}$$

The first objective function (33.62) is maximizing the total flow, and second objective function (33.63) is minimizing the total cost. Constraints (33.64) are called the flow conservation or Kirchhoff equations and indicate that the flow may be neither created nor destroyed in the network. In the conservation equations, sum of x_{jk} represents the total flow out of node j , while sum of x_{ij} indicates the total flow into node j . These equations require that the net flow out of node j should equal f . If $f < 0$, then there should be more flow into j than out of j .

Genetic Representation

In this problem, we consider a priority-based encoding method with special decoding for various network design problems. Cheng and Gen proposed priority-based genetic algorithm (priGA) firstly for solving resource-constrained project scheduling problem (rcPSP) [27]. The priority-based encoding method is an indirect approach. As it is known, a gene in a chromosome is characterized by two factors: locus, i.e., the position of gene located within the structure of chromosome, and allele, i.e., the value the gene takes. An example of priority-based encoding is shown in Fig. 33.23. For designing the decoding method, we first design a one-path growth procedure that obtains a path base on the generated chromosome with given network. We then design an overall-path growth procedure that removes the used flow from each arc and deletes the arcs whose capacity is 0. Table 33.1 shows two examples of the solution of bDN with various paths.

Table 33.1 Examples of the solutions with various paths

Solution	# of paths K	Path $P(\cdot)$	Total flow z_1	Total cost z_2
1	3	(1-3-6-5-8-11), (1-3-6-8-11), (1-3-7-6-9-11)	60	4660
2	8	(1-3-6-5-8-11), (1-3-6-8-11), (1-3-7-6-9-11), (1-4-7-6-9-11), (1-4-7-10-11), (1-2-5-8-11), (1-2-5-8-9-11), (1-2-6-9-10-11)	160	12450

Genetic Operators

For priority-based representation as a permutation representation, several crossover operators have been proposed, such as partial-mapped crossover (PMX), order crossover (OX), cycle crossover (CX), position-based crossover (PX), heuristic crossover, etc. [4]. In this BND model, we propose a weight mapping crossover (WMX); it can be viewed as an extension of one-cut-point crossover for permutation representation. At one-cut-point crossover, two chromosomes (parents) would choose a random cut-point and generate the offspring by using a segment of its own parent to the left of the cut-point and then remap the right segment based on the weight of other parent of the right segment. An example of the WMX is given in Fig. 33.24. For the mutation, we examine several kinds of mutation operators, and effectiveness of insertion mutation is the best one for priority-based representation. For the selection, it is mainly used to adjust genetic search in favor of a wide exploration of the search space when solving BND problems by the weighted-sum approach. We adopted roulette wheel selection that is the best selection type as supplementary to the weighted-sum approach.

In the BND model, we propose a fitness assignment approach by an interactive adaptive-weight GA (i-AWGA), which is an improved adaptive-weight fitness assignment approach with the consideration of the disadvantages of weighted-sum approach and Pareto ranking-based approach. We combine a penalty term to the fitness value for all of dominated solutions. Calculate the adaptive weight $w_1 = 1/(f_1^{\max} - f_1^{\min})$ for objective 1 and the adaptive weight $w_2 = 1/(f_2^{\max} - f_2^{\min})$ for objective 2. Afterward, calculate the penalty term $p(v_k) = 0$, if v_k is nondominated solution in the nondominated set P . Otherwise $p(v_k) = 1$ for dominated solution v_k . Last, calculate the fitness value of each chromosome by combining the method as follows, and we adopted roulette wheel selection as supplementary to the i-AWGA fitness assignment approach based on the following equation:

$$eval(v_k) = w_1(f_1^k - f_1^{\min}) + w_2(f_2^k - f_2^{\min}) + p(v_k), \quad \forall k \in popSize \tag{33.67}$$

Performance Measures

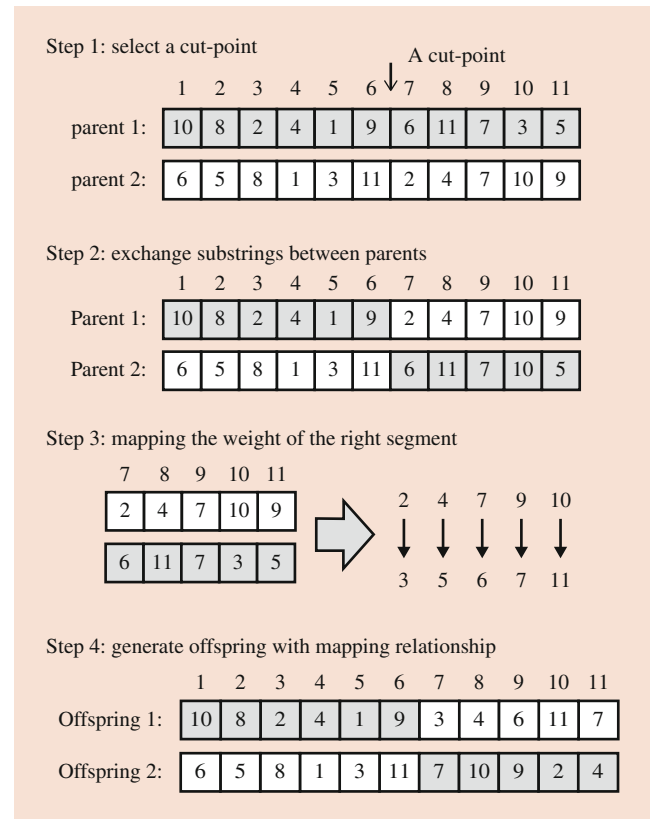


Fig. 33.24 An example of WMX

In order to evaluate the results of each test for the bicriteria network design problem, we are using the performance measures: average of the best solutions (ABS), percent deviation (PD) from optimal solution, and standard deviation (SD). We also give a statistical analysis by ANOVA and give examples of Pareto frontier, convergence patterns for the problems:

- (1) The number of obtained solutions $|S_j|$.
- (2) The ratio of nondominated solutions $R_{NDS}(S_j)$: A straightforward performance measure of the S_j with respect to the J solution sets is the ratio of solutions in S_j that are not dominated by any other solutions in S . The $R_{NDS}(S_j)$ measure can be written as follows:

Table 33.2 Fitness assignment approaches for the 25/49 test problem

# of eval. solut.	S _j				R _{NDS} (S _j)				D1 _R (S _j)			
	spEA	nsGAll	rwGA	i-awGA	spEA	nsGAll	rwGA	i-awGA	spEA	nsGAll	rwGA	i-awGA
50	41.20	43.60	42.60	44.00	0.35	0.33	0.34	0.33	181.69	180.64	168.73	168.96
500	49.80	56.60	51.60	57.50	0.47	0.50	0.42	0.46	104.77	114.62	119.53	103.13
2000	62.90	62.90	55.30	64.70	0.61	0.65	0.51	0.65	74.76	81.24	95.70	76.41
5000	67.80	68.40	60.70	69.40	0.73	0.72	0.64	0.73	62.97	62.77	80.68	62.33

Table 33.3 Comparison results using the three performance measures

Test problems (# of nodes/ # of arcs)	S _j		R _{NDS} (S _j)		D1 _R	
	pri-awGA	mo-hGA	pri-awGA	mo-hGA	pri-awGA	mo-hGA
25/49	60	58	0.379	0.483	1.945	0.727
25/56	67	60	0.516	0.597	1.647	1.207

$$R_{NDS}(S_j) = \frac{|S_j\{x \in S_j | \exists r \in S^* : r < x\}|}{|S_j|} \quad (33.68)$$

(3) The distance D1_R measure can be written as follows:

$$D1_R = \frac{1}{|S^*|} \sum_{r \in S^*} \min\{d_{rx} | x_t \in S_j\} \quad (33.69)$$

where S* is a reference solution set for evaluation of the solution set S_j and d_{rx} is the distance between a current solution x and a reference solution r.

$$d_{rx} = \sqrt{(f_1(r) - f_1(x))^2 + (f_2(r) - f_2(x))^2} \quad (33.70)$$

Experimental Result

We compare i-AWGA with SPEA, NSGA II, and rwGA through computational experiments under the same stopping condition (i.e., evaluation of 5000 solutions). Each algorithm was applied to each test problem 10 times and gives the average results of the three performance measures. In Table 33.2, better results of |S_j| and D1_R were obtained by the i-AWGA than other fitness assignment approach. As shown in Table 33.3, the Mo-HGA got the shortest distance D1_R and also gives better performance than pri-AWGA by R_{NDS}(S_j) measure. However, Mo-HGA did not effectively combine the number of obtained solutions |S_j|. We show the Pareto optimal solutions obtained from pri-AWGA and Mo-HGA with the test problem comprised of 25 nodes and 49 arcs in Fig. 33.25.

33.4 Scheduling Problem

Scheduling problems exist almost everywhere in real-world situations, especially in the industrial engineering world. Many scheduling problems from manufacturing industries are quite complex in nature and very difficult to solve by conventional optimization techniques [35].

33.4.1 Flow-Shop Sequencing Model

The flow-shop sequencing problem is generally described as follows: there are *m* machines and *n* jobs, each job consists of *m* operations, and each operation requires a different machine. *n* jobs have to be processed in the same sequence on *m* machines. The processing time of job *i* on machine *j* is given by *t_{ij}* (*i* = 1, ..., *n*; *j* = 1, ..., *m*). The objective is to find the sequence of jobs minimizing the maximum-flow time, which is called makespan.

Heuristics for general *m*-machine problems: i.e., genetic algorithms have been successfully applied to solve flow-shop problems. We describe Gen, Tsujimura, and Kubota's approach [36].

Representation Because the flow-shop problem is essentially a permutation schedule problem [4, 37, 38], we can use the permutation of jobs as the representation scheme of chromosome, which is the natural representation for a sequencing problem. For example, let the *k*-th chromosome be

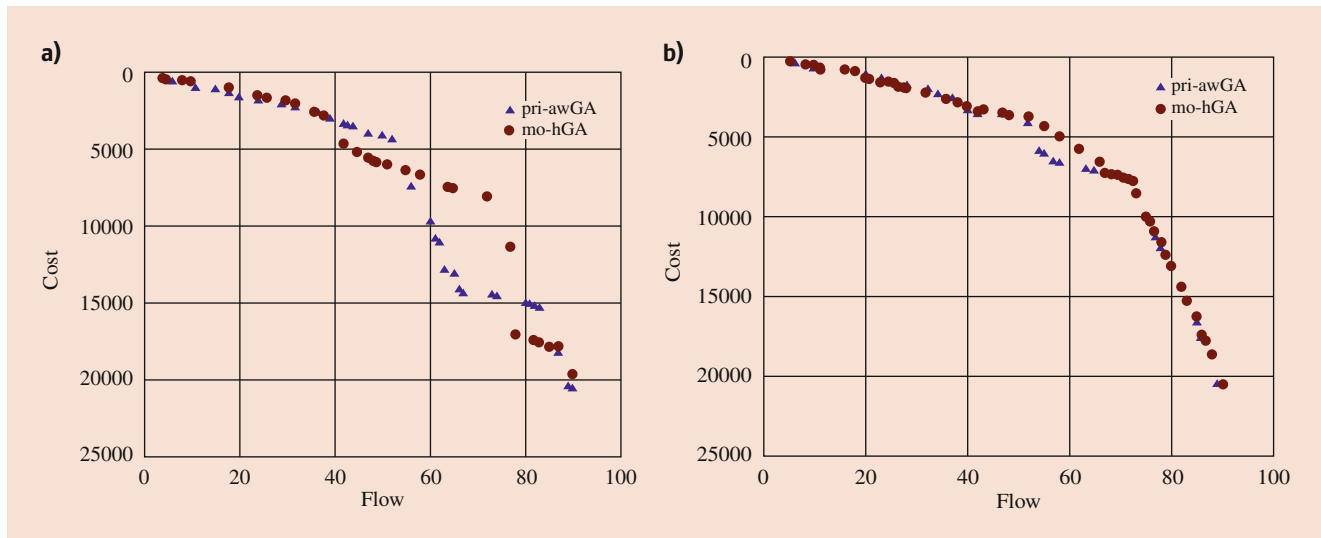


Fig. 33.25 Pareto optimal solutions by pro-AWGA and Mo-HGA with the 25/49 test problem. (a) The result is obtained at generation $gen = 50$. (b) The result is obtained at generation $gen = 100$

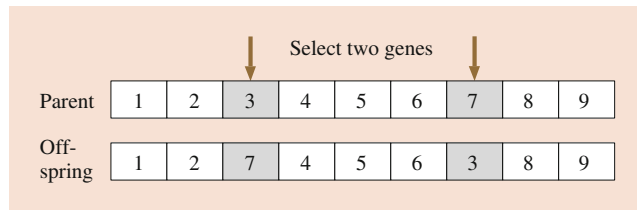


Fig. 33.26 Swap mutation

$$v_k = [3\ 2\ 4\ 1]$$

meaning that the jobs sequence is j_3, j_2, j_4, j_1 .

Crossover and Mutation Here, Goldberg’s PMX is used. Mutation is designed to perform random exchange; that is, it selects two genes randomly in a chromosome and exchanges their positions. An example is given in Fig. 33.26.

33.4.2 Job-Shop Scheduling Model

In the job-shop scheduling problem (JSP), we are given a set of jobs and a set of machines. Each machine can handle at most one job at a time. Each job consists of a chain of operations, each of which needs to be processed during an uninterrupted time period of a given length on a given machine. The purpose is to find a schedule, that is, an allocation of the operations to time intervals on the machines, which has a minimum duration required to complete all jobs [36,39,40].

Adapted Genetic Operators

During the past two decades, various crossover operators have been proposed for literal permutation encodings, such

as partial-mapped crossover (PMX), order crossover (OX), cycle crossover (CX), etc.

PMX is explained in the previous section.

Order Crossover (OX): Order crossover was proposed by Davis. OX has the following major steps [4, 25]:

- Step 1. Select a substring from one parent at random.
- Step 2. Produce a proto-child by copying the substring into the corresponding positions as they are in the parent.
- Step 3. Delete all the symbols from the second parent that are already in the substring. The resulted sequence contains the symbols the proto-child needs.
- Step 4. Place the symbols into the unfixed positions of the proto-child from left to right according to the order of the sequence to produce an offspring.

Cycle Crossover (CX): Cycle crossover was proposed by Oliver et al. CX works as follows:

- Step 1 Find the cycle which is defined by the corresponding positions of symbols between parents.
- Step 2 Copy the symbols in the cycle to a child with the corresponding positions of one parent.
- Step 3 Determine the remaining symbols for the child by deleting those symbols which are already in the cycle from the other parent.
- Step 4 Fill the child with the remaining symbols.

Mutation: It is relatively easy to make some mutation operators for the permutation representation. During the last decade, several mutation operators have been proposed for permutation representation, such as inversion, insertion, displacement, reciprocal exchange mutation, and shift mutation.

Inversion mutation selects two positions within a chromosome at random and then inverts the substring between these two positions. *Insertion mutation* selects a gene at random and inserts it in a random position.

33.4.3 Flexible Job-Shop Scheduling Model

Flexible job-shop scheduling problem (FJSP) is an extension of the traditional job-shop scheduling problem, which provides a closer approximation to real scheduling problems [41–44]. In the job-shop scheduling problem (JSP), there are n jobs that must be processed on a group of m machines. Each job i consists of a sequence of n_i operations ($o_{i1}, o_{i2}, \dots, o_{in_i}$), where o_{ik} (the k -th operation of job i) must be processed without interruption on a predefined machine m_{ik} for p_{ik} time units. The operations $o_{i1}, o_{i2}, \dots, o_{in_i}$ must be processed one after another in the given order, and each machine can process at most one operation at a time.

The first objective function accounts for makespan, Eq. (33.71) combining with Eq. (33.72) giving a physical meaning to the FJSP, referring to reducing total processing time and dispatching the operations averagely for each machine.

Mathematical Model of FJSP

$$\min c_M = \max_{1 \leq i \leq n} \{c_{in_i}\} \tag{33.71}$$

$$\min w_M = \max_{1 \leq j \leq m} \left\{ \sum_{i=1}^n \sum_{k=1}^{n_i} t_{ikj} x_{ikj} \right\} \tag{33.72}$$

$$\min w_T = \sum_{i=1}^n \sum_{k=1}^{n_i} \sum_{j=1}^m t_{ikj} x_{ikj} \tag{33.73}$$

$$\text{s. t. } c_{ik} - c_{i(k-1)} \geq t_{ikj} x_{ikj}, \quad k = 2, \dots, n_i; \quad \forall i, j \tag{33.74}$$

$$[(c_{hg} - c_{ik} - t_{hgi})x_{hgi}x_{ikj} \geq 0] \vee [(c_{ik} - c_{hg} - t_{ikj})x_{hgi}x_{ikj} \geq 0], \quad \forall (i, k), (h, g), j \tag{33.75}$$

$$\sum_{j \in A_{ik}} x_{ikj} = 1, \quad \forall i, k \tag{33.76}$$

$$x_{ikj} \in \{0, 1\}, \quad \forall i, k, j \tag{33.77}$$

$$c_{ik} > 0, \quad \forall i, k \tag{33.78}$$

function Eq. (33.73) is to minimize the total workload. Considering both equations, our objective is to balance the workloads of all machines. Inequality (33.74) describes the operation precedence constraints. Inequality (33.75) is a disjunctive constraint, where one or the other constraint must be observed.

It represents that the operation o_{hg} should not be started before the completion of operation o_{ik} or that the operation

o_{hg} must be completed before the starting of operation o_{ik} , if they are assigned on the same machine j . Shortly, the execution of operation o_{ik} cannot be overlapped in time with the execution of operation o_{hg} , which describes the operation precedence constraints. Equation (33.76) states that one machine must be selected from a set of available machines for each operation. Equations (33.77–33.78) are variable restrictions on decision variables [45, 46].

Multistage Operation-Based GA

For saving CPU time on several representations, a multistage operation-based GA approach has been proposed [47]. The two-vector multistage operation-based genetic algorithm (MoGA) for the FJSP is as follows:

Phase 1: Creating an operation sequence

- Step 1.1: Generate a random priority to each operation in the model using priority-based encoding procedure for the vector v_1 .
- Step 1.2: Decode a feasible operation sequence that satisfies the precedence constraints of FJSP by the priority-based decoding procedure.

Phase 2: Assigning operations to machines

- Step 2.1: Input the operation sequence found in step 1.2.
- Step 2.2: Generate a permutation encoding for machine assignment of each operation (second vector v_2).

Phase 3: Designing a schedule

- Step 3.1: Create a schedule S using operation sequence and machine assignments.
- Step 3.2: Draw a Gantt chart for the schedule.

For introducing the multistage operation-based GA for solving a simple example of the FJSP model, we first prepare the data set including three jobs operated on four machines in Table 33.4.

Table 33.4 Data set of a three-job four-machine problem

		M_1	M_2	M_3	M_4
J_1	O_{11}	1	3	4	1
	O_{12}	3	8	2	1
	O_{13}	3	5	4	7
J_2	O_{21}	4	1	1	4
	O_{22}	2	3	9	3
	O_{23}	9	1	2	2
J_3	O_{31}	8	6	3	5
	O_{32}	4	5	8	1

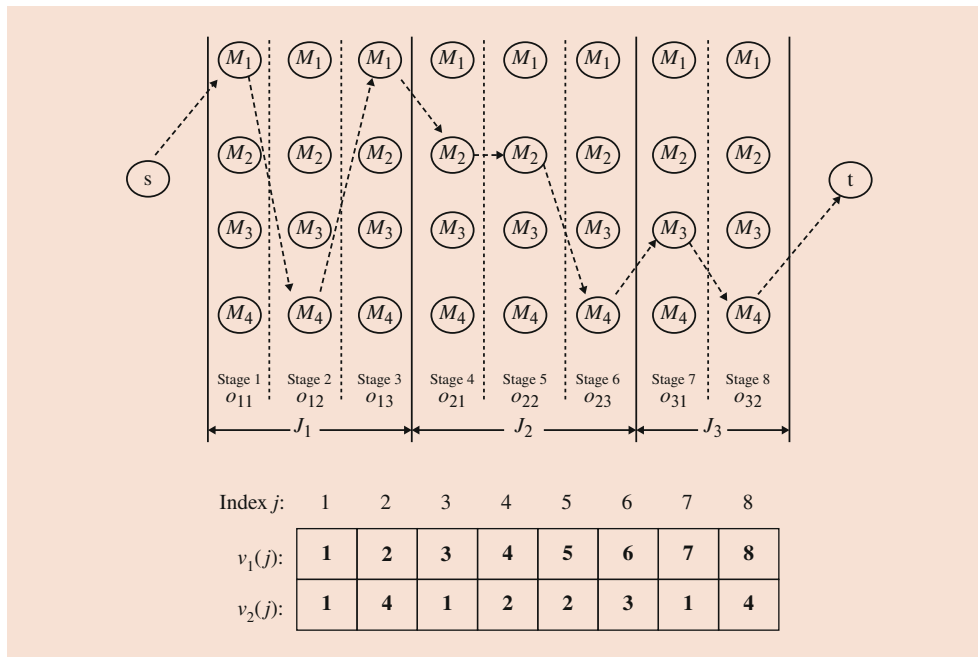


Fig. 33.27 Multistage operation-based representation of simple FJSP example

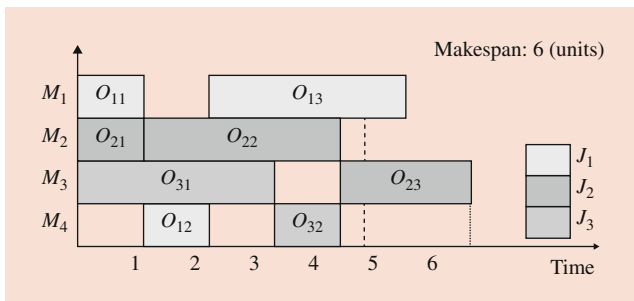


Fig. 33.28 Gantt chart for simple FJSP example

For the multistage operation-based representation of the simple FJSP example shown in Table 33.4, we denote each operation as one stage and each machine as one state; the problem can be formulated into an eight-stage, four-state problem; and also connected by the dashed arcs, a feasible schedule can be obtained as Fig. 33.27.

After Phases 1 and 2 of the multistage operation-based GA, we can create a schedule S using operation sequence and machine assignments as follows:

$$\begin{aligned}
 S &= \{(o_{11}, M_1), (o_{12}, M_4), (o_{13}, M_1), (o_{21}, M_2), \\
 & \quad (o_{22}, M_2), (o_{23}, M_1), (o_{31}, M_3), (o_{32}, M_4)\} \\
 &= \{(o_{11}, M_1 : 0 - 1), (o_{11}, M_4 : 1 - 2), (o_{11}, M_1 : \\
 & \quad 2 - 5), (o_{11}, M_2 : 0 - 1), (o_{11}, M_2 : 1 - 4), (o_{11}, \\
 & \quad M_1 : 4 - 6), (o_{11}, M_3 : 1 - 3), (o_{11}, M_4 : 3 - 4)\}
 \end{aligned}$$

From this schedule S , we can draw the Gantt chart as shown in Fig. 33.28.

The main advantages of the two-vector representation are that each possible chromosome always represents a feasible solution candidate and that the coding space is smaller than that of permutation representation. For the detailed solving process of the FJSP such as the priority-based decoding and neighborhood search method using critical path, we can refer to Section 5.3 Flexible Job-Shop Scheduling Model in [48–51].

Fitness Function

The three considered objective functions do not conflict with one another as seriously as in most other multiobjective optimization problems, because a small makespan (c_M) requires a small maximal workload (w_M) and a small maximal workload implies a small total workload (w_T). During evaluation, the fitness of a solution is calculated by synthesizing the three objectives into a weighted sum. We have to normalize the objective values on the three criteria before they are summed since they are of different scales. Let $c_M(\mathbf{v})$ be the makespan of the chromosome. The scaled makespan ($c'_M(\mathbf{v})$) of a solution \mathbf{v} is as follows:

$$c'_M(\mathbf{v}) = \begin{cases} \frac{c_M(\mathbf{v}) - c_M^{\min}}{c_M^{\max} - c_M^{\min}}, & \text{if } c_M^{\max} \neq c_M^{\min}, \text{ for all } \mathbf{v} \\ 0.5 & \text{otherwise,} \end{cases} \tag{33.79}$$

where

$$c_M^{\min} = \min_{1 \leq l \leq p} \{c_M(\mathbf{v})\} \quad (33.80)$$

$$c_M^{\max} = \max_{1 \leq l \leq p} \{c_M(\mathbf{v})\} \quad (33.81)$$

After scaling, the three objectives all take values from the range of [0, 1]. In order to guide the genetic and local search to the most promising area, makespan is given a very large weight since the other two objectives heavily depend on it. Additionally, it is typically the most important criterion in practical production environments. For the FJSP problem, a number of solutions with different maximal workloads or total workloads may have the same makespan. From this point of view, we firstly find the solutions with the minimum makespan and then minimize the maximal workload and the total workload in the presence of the minimum makespan. The fitness of a solution \mathbf{v} then is

$$eval(\mathbf{v}) = \alpha_1 \cdot c'_M(\mathbf{v}) + \alpha_2 \cdot w'_M(\mathbf{v}) + \alpha_3 \cdot w'_T(\mathbf{v})$$

$$\text{where, } \alpha_1 > \alpha_2 > \alpha_3 > 0 \text{ and } \alpha_1 + \alpha_2 + \alpha_3 = 1. \quad (33.82)$$

33.4.4 Resource-Constrained Project Scheduling Model

The problem of scheduling activities under resource and precedence restrictions with the objective of minimizing the project duration is referred to as the resource-constrained project scheduling problem in the literature [27, 52].

The problem can be stated mathematically as follows:

$$\min t_n, \quad (33.83)$$

$$\text{s. t. } t_j - t_i \geq d_i, \quad \forall j \in S_i, \quad (33.84)$$

$$\sum_{t_i \in A_{ik}} r_{ik} \leq b_k, \quad k = 1, 2, \dots, m, \quad (33.85)$$

$$t_i \geq 0, \quad i = 2, \dots, n, \quad (33.86)$$

where t_i is the starting time of activity i , d_i the duration (processing time) of activity i , S_i the set of successors of activity i , r_{ik} the amount of resource k required by activity i , b_k the total availability of resource k , A_{ik} the set of activities in process at time t_i , and m the number of different resource types. Activities 1 and n_4 are dummy activities which mark the beginning and end of the project. The objective is to minimize the total project duration.

A. Priority-Based Encoding

For this problem, priority-based encoding is used; it is explained in the previous section.

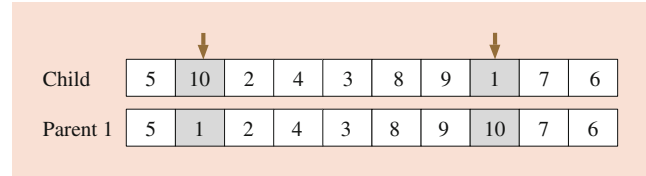


Fig. 33.29 Swap mutation operator

B. Genetic Operators

Position-Based Crossover: The position-based crossover operator is used. This crossover is explained in the previous section.

Swap Mutation: The swap mutation operator was used here, which simply selects two positions at random and swaps their contents, as shown in Fig. 33.29.

33.4.5 Multiprocessor Scheduling Model

The multiprocessor scheduling is to assign n tasks to m processors in such a way that precedence constraints are maintained and to determine the start and finish times of each task with the objective of minimizing the completion time. There is a paper which deals with real-time tasks [4, 53]. However, here we introduce an algorithm concerned with general tasks. The mathematical formulation of the problem is given as

$$\min[\max_j(x_j y_{ij})] \quad (33.87)$$

$$\text{s. t. } x_k - x_j \geq P_k, T_j < T_k, \quad (33.88)$$

$$\sum_{j=1}^n p_j y_{ij}, \quad j = 1, \dots, n, \quad (33.89)$$

$$\sum_{j=1}^m y_{ij} = 1, \quad j = 1, \dots, n, \quad (33.90)$$

$$y_{ij} = 0 \text{ or } 1, \quad i = 1, \dots, n, j = 1, \dots, n, \quad (33.91)$$

where

$$x_{ij} = \begin{cases} 1, & \text{if task } T_j \text{ is assigned to processor } P_i \\ 0, & \text{otherwise.} \end{cases} \quad (33.92)$$

and where $t_{\max} = \max_i(t_i)$, x_j is the completion time of task T_j , p_j is the processing time of task T_j , t_i is the time required to process all tasks assigned to process P_i , and $<$ represents a precedence relation; a precedence relation between tasks, $T_j < T_k$, means that T_k precedes T_j .

Genetic Algorithm for MSP

For the chromosome representation scheme and genetic operations, we adopt the concept of the height function, which considers precedence relations among the tasks in the implementation of a genetic algorithm.

Height Function To facilitate the generation of the schedule and the construction of the genetic operators, we define the *height* of each task in the task graph as

$$\text{height}(T_i) = \begin{cases} 0, & \text{if } \text{pre}(T_i) = \phi \\ 1 + \max_{T_k \in \text{pre}(T_i)} [\text{height}(T_k)], & \text{otherwise.} \end{cases} \quad (33.93)$$

$$\begin{aligned} \text{height}'(T_j) &= \text{rand} \in \{\max[\text{height}(T_i)] + 1\}, \\ \min[\text{height}'(T_k)] &\text{ over all } T_i \in \text{pre}(T_j) \text{ and} \\ T_k &\in \text{suc}(T_j) \end{aligned} \quad (33.94)$$

where $\text{pre}(T_j)$ is the set of predecessors of T_j and $\text{suc}(T_j)$ is the set of successors of T_j .

Representation The chromosome representation used here is based on the schedule of the tasks in each processor. The representation of the schedule for genetic algorithms must accommodate the precedence relations between the computational tasks.

Genetic Operators The function of the genetic operators is to create new search nodes based on the current population of search nodes. New search nodes are typically constructed by combining or rearranging parts of the old search nodes.

Operation 1: Operation 1 is performed in the following steps:

Step 1 Generate a random number c from the range $[1, \max(\text{height}')]]$.

Step 2 Place the cut-point at each processor in such a way that the tasks' height' before the cut-point is less than c and more than or equal to c after the cut-point.

Step 3 Exchange the second partial schedules.

Operation 2: Operation 2 is performed in the following steps:

Step 1 Generate a random number c from the range $[1, \max(\text{height}')]]$

Step 2 At each processor, pick all tasks whose height' is c .

Step 3 Replace the position of all tasks randomly.

33.4.6 Assembly Line Balancing

An *assembly line* (AL) is a manufacturing process consisting of various tasks in which interchangeable parts are added to a product in a sequential manner at a station to produce a finished product. Most of the works related to the ALs concentrate on the *assembly – line balancing* (ALB). The ALB model deals with the allocation of the tasks among stations so that the precedence relations are not violated and a given objective function is optimized [54,55].

Based on the model structure, ALB models can be classified into two groups. The first group [55] includes single-model *assembly line balancing* (smALB), multi-model *assembly line balancing* (muALB), and mixed-model *assembly line balancing* (mALB); the second group includes *simple assembly linebalancing* (sALB) and general *assembly line balancing* (gALB) [56–60].

Mathematical Model

For formulating a mathematical programming model, firstly we define the following utilization and average utilization of the station, respectively:

$$u_i = \frac{1}{\max_{1 \leq i \leq m} \{t(S_i)\}} t(S_i) \quad (33.95)$$

$$\bar{u} = \frac{1}{m} \sum_{i=1}^m u_i \quad (33.96)$$

The mathematical model for the simple assembly line balancing (sALB) Type 1 can be stated as the 0-1 as follows:

$$\max E = \frac{1}{m c_T} \sum_{j \in S_i} t_j x_{ij} \quad (33.97)$$

$$\min m = \sum_{i=1}^M \max_{1 \leq j \leq x} \{x_{ij}\} \quad (33.98)$$

$$\min V = \sqrt{\frac{1}{m} \sum_{i=1}^m (u_i - \bar{u})^2} \quad (33.99)$$

$$\text{s. t. } \sum_{i=1}^M x_{ij} = 1, \forall j \quad (33.100)$$

$$\sum_{i=1}^M i x_{ik} \leq \sum_{i=1}^M i x_{ij}, \forall j, \forall k \in \text{pre}(j) \quad (33.101)$$

$$t(S_j) = \sum_{j \in S_i} t_j = \sum_{j=1}^M t_j x_{ij} \leq c_T, \forall i \quad (33.102)$$

$$x_{ij} = 0 \text{ or } 1, \quad \forall i, j. \quad (33.103)$$

M: maximum number of stations available ($n \leq M$). In this mathematical model, the first objective Eq. (33.101) is to maximize the line efficiency. The second objective Eq. (33.102) is to minimize the number of stations actually employed. The third objective Eq. (33.103) is to minimize the variation of workload. The constraints given in Eqs. (33.104–33.106) are used to formulate the general feasibility of the problem. The constraint (33.104) states that each task must be assigned to one and only one station. The inequality (33.105) represents the precedence constraints, and it states that the direct predecessor of task j must be assigned to a station which is in front of or the same as the station that task j is assigned in. This constraint stresses that if a task is assigned to a station, then the predecessor of this task must be already assigned to a station. The inequality (33.106) denotes that the available time at each station should be less than or equal to the given cycle time. Constraint (33.107) represents the usual integrity restriction.

Priority-Based GA

In this section, we will introduce a priority-based genetic algorithm (priGA) for solving the multiobjective sALB Type 1 model. The priGA approach was originally developed by Gen and Cheng [10] in order to handle the problem of creating encoding while treating the precedence constraints efficiently. For applying a priority-based genetic representation for the sALB model, there are three main phases:

Phase 1: Creating a task sequence

- Step 1.1: Generate a random priority to each task in the model using encoding procedure.
- Step 1.2: Decode a feasible task sequence T_S that satisfies the precedence constraints.

Phase 2: Assigning tasks to stations

- Step 2.1: Input the task sequence found in T_S
- Step 2.2: Obtain a feasible solution set according to this task sequence.

Phase 3: Designing a schedule

- Step 3.1: Create a schedule S using station assignments found in step 2.2.
- Step 3.2: Draw a Gantt chart for this schedule.

Table 33.5 presents the data set for an example sALB model, which contains 12 tasks. Using this data set, the precedence graph in Fig. 33.30 is constructed.

The precedence graph contains 12 nodes for tasks, node weights for task processing times, and arcs for orderings. Phase 1: Creating a Task Sequence

By using the priority-based encoding, the position of a gene was used to represent a task node, and the value of the gene was used to represent the priority of the task node

Table 33.5 Data set of the sALB model

Task j	Sue(j)	Task time t_j
1	{2,4}	5
2	{3}	3
3	{6}	4
4	{5}	3
5	{6}	6
6	{7,9,10}	5
7	{8}	2
8	{12}	6
9	{12}	1
10	{11}	4
11	{12}	4
12	{}	7

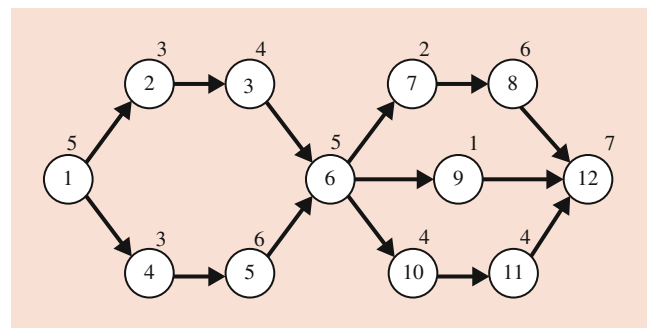


Fig. 33.30 Precedence graph of the sALB model

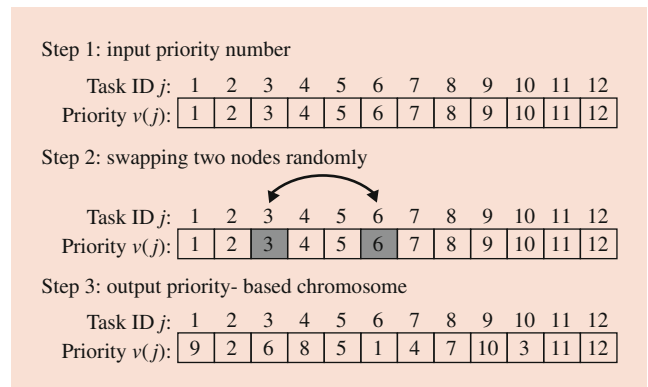


Fig. 33.31 Illustration of an example priority-based encoding

for constructing a schedule among candidates. This encoding method verifies any permutation-type representations, so that most of the existing genetic operators can be easily applied. Figure 33.31 illustrates the process of this encoding procedure on a chromosome.

Then the next possible nodes are 2 and 5. They have priority of 2 and 5, respectively, and then we put 5 into task sequence T_S . Finally, we repeat these steps until we obtain a complete schedule $T_S = \{1, 4, 5, 2, 3, 6, 9, 7, 8, 10, 11, 12\}$. Phase 2: Assigning Tasks to Stations

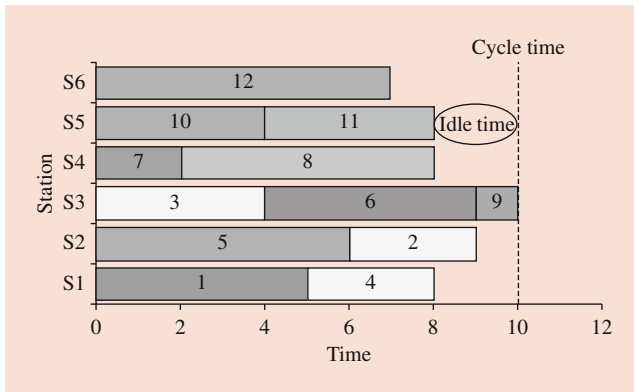


Fig. 33.32 Gantt chart for the sALB model (one unit of product)

In this phase, the assignments of task to stations are formed using the task sequence found in step 1.2. For the illustration of this decoding procedure, the task sequence $T_S = \{1, 4, 5, 2, 3, 6, 9, 7, 8, 10, 11, 12\}$ found in step 1.2 is used. Table 33.6 presents the trace table for the decoding procedure. In the example, we obtained a feasible line balance with cycle time 10 time units and 6 stations represented by the station loads $S_1 = 1, 4, S_2 = 2, 5, S_3 = 3, 6, 9, S_4 = 7, 8, S_5 = 10, 11, S_6 = 12$. While no idle time occurs in station 3, stations 1, 2, 4, 5, and 6 show idle times of 2, 1, 2, 2, and 3 time units, respectively.

Phase 3: Designing a Schedule

For creating a schedule S , we can use the station assignment found in Step 2.2. Using the trace table for the decoding procedure, the schedule can be constructed as follows:

Schedule $S = \{(j, S_i, t_j)\}$:

$$\begin{aligned}
 S = & \{(1, S_1, t_1), (4, S_1, t_4), (5, S_2, t_5), (2, S_2, t_2), \\
 & (3, S_1, t_3), (6, S_1, t_8), (9, S_1, t_9), (7, S_1, t_7), \\
 & (8, S_1, t_8), (10, S_1, t_{10}), (6, S_1, t_8), (9, S_1, t_9), \\
 & (7, S_1, t_7), (8, S_1, t_8), (10, S_1, t_{10}), (11, S_1, t_{11}), \\
 & (12, S_1, t_{12})\} \\
 = & \{(1, S_1, 0 - 5), (4, S_1, 5 - 8), (5, S_1, 8 - 14), \\
 & (2, S_1, 14 - 17), (3, S_1, 17 - 21), (6, S_1, 21 - 26), \\
 & (9, S_1, 26 - 27), (7, S_1, 27 - 29), (8, S_1, 29 - 35), \\
 & (10, S_1, 35 - 39), (11, S_1, 39 - 43), \\
 & (12, S_1, 43 - 50)\}
 \end{aligned}$$

By using the schedule S , we can draw a Gantt chart as shown in Fig. 33.32 which illustrates a feasible schedule for one unit of product. However, since a single type of product is produced on a simple assembly line system, the minimization of the makespan becomes an important production management issue.

Table 33.6 Trace table for task to station assignment decoding procedure

j	\bar{s}	$v(j)$	$j^*(t_j)$	$S_j = \{ \}; t(S_j) (c_T - t(S_j))$
0	{1}	$v(1)=9$	1(5)	$S_j = \{1\}; 5 (5)$
1	{2,4}	$v(2)=2, v(4)=8$	4(3)	$S_j = \{1,4\}; 8 (2)$
4	{2,5}	$v(2)=2, v(5)=5$	5(6)	$S_2 = \{5\}; 6 (4)$
5	{2}	$v(3)=6$	2(3)	$S_2 = \{5,2\}; 9 (1)$
2	{3}	$v(3)=6$	3(4)	$S_3 = \{3\}; 4 (6)$
3	{6}	$v(6)=1$	6(5)	$S_3 = \{6\}; 9 (1)$
6	{7,9,10}	$v(7)=4, v(9)=10, v(10)=3$	9(1)	$S_3 = \{3,6,9\}; 10 (0)$
9	{7,10}	$v(7)=4, v(10)=3$	7(2)	$S_4 = \{7\}; 2 (8)$
7	{8,10}	$v(8)=7, v(10)=3$	8(6)	$S_4 = \{7,8\}; 8 (2)$
8	{10}	$v(10)=3$	10(4)	$S_5 = \{10\}; 4 (6)$
10	{11}	$v(11)=11$	11(4)	$S_5 = \{10,11\}; 8 (2)$
11	{12}	$v(12)=12$	12(7)	$S_6 = \{12\}; 7 (3)$

This time, Fig. 33.33 shows a Gantt chart for three units of product in which the makespan is 70 time units for three units of products.

In this study, adaptive-weight approach that utilizes some useful information from the current population to readjust weights for obtaining a search pressure toward a positive ideal point I is used. Two objective functions, i.e., maximization of line efficiency (Eq. 42.97) and minimization of variation of workload (Eq. 42.99), are used in this study:

$$\begin{aligned}
 eval(v_k) &= w_1(f_1(v_k) - z_1^{min}) + w_3(f_3(v_k) - z_3^{min}) \\
 &= w_1 \left\{ \frac{1}{mc_T} \sum_{j=1}^n t_j - z_1^{min} \right\} + w_3 \\
 &\quad \left\{ 1 - \sqrt{\frac{1}{m} \sum_{i=1}^m (u_i - \bar{u})^2} - z_3^{min} \right\} \\
 k &= 1, \dots, popSize
 \end{aligned} \tag{33.104}$$

33.5 Reliability Design Problem

Reliability optimization appeared in the late 1940s and was first applied to communication and transportation systems. Much of the early work was confined to the analysis of certain performance aspects of systems. One goal of the reliability engineer is to find the best way to increase system reliability.

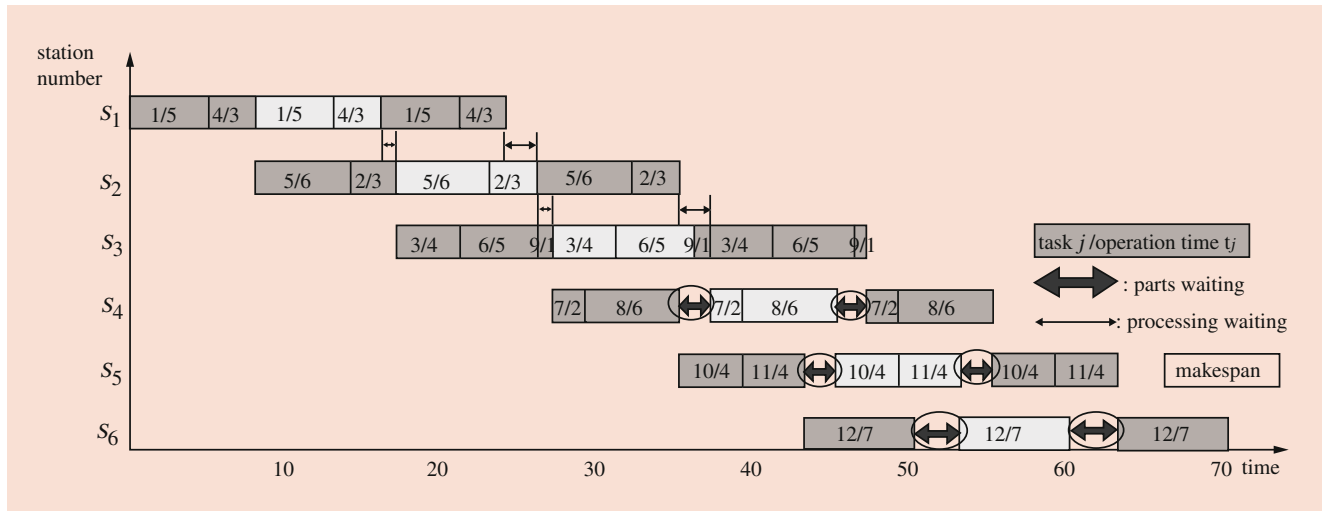


Fig. 33.33 Gantt chart for the sALB model (three units of product)

The reliability of a system can be defined as the probability that the system has operated successfully over a specified interval of time under stated conditions [61].

33.5.1 Genetic Algorithm for Reliability Optimization

The problem is to maximize the system reliability subject to three nonlinear constraints with parallel redundant units in subsystems that are subject to A failures, which occur when the entire subsystem is subjected to the failure condition. It can be mathematically stated as follows:

$$\max R(\mathbf{m}) = \prod_{i=1}^3 \{1 - [1 - (1 - q_{iu})^{m_i+1}] - \sum_{n=2}^4 (q_{iu})^{m_i+1}\}, \quad (33.105)$$

$$\text{s. t. } G_1(\mathbf{m}) = (m_1 + 3)^2 + (m_2)^2 + (m_3)^2 \leq 51, \quad (33.106)$$

$$G_2(\mathbf{m}) = 20 \sum_{i=1}^3 [m_i + \exp(-m_i)] \geq 120 \quad (33.107)$$

$$G_3(\mathbf{m}) = 20 \sum_{i=1}^3 [m_i \exp(-m_i/4)] \geq 65 \quad (33.108)$$

$$1 \leq m_1 \leq 4, \quad 1 \leq m_2, m_3 \leq 7 \quad (33.109)$$

$$m_i \geq 0; \text{ Integer, } i = 1, 2, 3 \quad (33.110)$$

Table 33.7 Failure modes and probabilities in each

Subsystem i	Failure modes $s_i = 4, h_i = 1$	Failure probability q_{iu}
1	O	0.01
	A	0.05
	A	0.10
	A	0.18
2	O	0.08
	A	0.02
	A	0.15
	A	0.12
3	O	0.04
	A	0.05
	A	0.20
	A	0.10

where $\mathbf{m} = (m_1 \ m_2 \ m_3)$. The subsystems are subject to four failure modes ($s_i = 4$) with one O failure ($h_i = 1$) and three A failures, for $i = 1, 2, 3$. For each subsystem the failure probability is shown in Table 33.7.

Genetic Algorithm Approach

Representation: The integer value of each variable m_i is represented as a binary string. The length of subsystem the string depends on the upper bound u_i of the redundant units. For instance, when the upper bound u_i equals 4, we need three binary bits to represent m_i .

In this example, the upper bounds of the redundant units in each subsystem are $u_1 = 4, u_2 = 7, \text{ and } u_3 = 7$, so each decision variable m_i needs three binary bits. This means that a total of nine bits are required. If $m_1 = 2, m_2 = 3, \text{ and } m_3 = 3$, we have the following chromosome:

$$\begin{aligned} \mathbf{v} &= [x_{33} \ x_{32} \ x_{31} \ x_{23} \ x_{22} \ x_{21} \ x_{13} \ x_{12} \ x_{11}] \\ &= [0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0] \end{aligned}$$

where x_{ij} is the symbol for the j -th binary bit of variable m_i .
Crossover: One-cut-point crossover is used here.
Mutation: Mutation is performed on a bit-by-bit basis.

33.5.2 Reliability Design with Redundant Unit and Alternatives

Gen, Yokota, Ida, and Taguchi further extended their work to the reliability optimization problem by considering both redundant units and alternative design [30, 61, 62, 75]. The example used here was firstly given by Fyffe et al. as follows:

$$\max R(\mathbf{m}, \boldsymbol{\alpha}) = \prod_{i=1}^{14} \{1 - [1 - R_i(\alpha_i)]^{m_i}\}, \quad (33.111)$$

$$\text{s. t. } G_1(\mathbf{m}, \boldsymbol{\alpha}) = \sum_{i=1}^{14} c_i(\alpha_i)m_i \leq 130, \quad (33.112)$$

$$G_2(\mathbf{m}, \boldsymbol{\alpha}) = \sum_{i=1}^{14} w_i(\alpha_i)m_i \leq 170, \quad (33.113)$$

$$1 \leq m_i \leq u_i, \quad \forall i, \quad (33.114)$$

$$1 \leq \alpha_i \leq \beta_i, \quad \forall i, \quad (33.115)$$

$$m_i, \alpha_i \geq 0; \quad \text{Integer, } \forall i, \quad (33.116)$$

where α_i represents the design alternative available for the i -th subsystem, m_i represents the identical units used in redundancy for the i -th subsystem, u_i is the upper bound of the redundant units for the i -th subsystem, and β_i is the upper bound of alternative design for the i -th subsystem.

Genetic Algorithm Approach

Representation The representation can be written as follows:

$$\mathbf{v} = [(\alpha_{k1}, m_{k1}) (\alpha_{k2}, m_{k2}) \dots (\alpha_{k14}, m_{k14})]$$

where α_{ki} is a design alternative, m_{ki} is a redundant unit, and the subscript k is the index of chromosome.

Crossover The uniform crossover operator given by Syswerda is used here, which has been shown to be superior to traditional crossover strategies for combinatorial problem. Uniform crossover firstly generates a random crossover mask and then exchanges relative genes between parents according to the mask. A crossover mask is simply a binary string with the same size of chromosome.

33.5.3 Network Reliability Design

A communication network can be represented by an undirected graph $G = (V, E)$, in which the nodes V and edges E represent computer sites and communication cables, respectively. The optimal design of network can be represented as follows [63]:

$$\min z(\mathbf{x}) \times 1 = \sum_{i=1}^{n-1} \sum_{j=i+1}^n c_{ij}x_{ij} \quad (33.117)$$

$$\text{s. t. } R(\mathbf{x}) \geq R_{\min} \quad (33.118)$$

Genetic Algorithm Approach

Representation: A genetic algorithm lends itself to this problem because each network design x is easily formed into a binary string which can be used as a chromosome for genetic algorithms. Each element of the chromosome represents a possible edge in the network design problem, so there are $n(n-1)/2$ string components in each candidate architecture Z .

Crossover: The one-cut-point crossover operation is used.

Mutation: The bit-flip mutation operation is employed, performed on a bit-by-bit basis.

33.5.4 Tree-Based Network Design

Consider a local-area network (LAN) that connects m users (stations). Also, we assume the $n \times m$ service center topology matrix \mathbf{x}_1 , which represents the connection between service centers. An element x_{1ij} is represented as

$$x_{1ij} = \begin{cases} 1, & \text{if the center } i \text{ and } j \text{ are connected} \\ 0, & \text{otherwise.} \end{cases} \quad (33.119)$$

Assume that the LAN is partitioned into n segments (service centers or clusters). The users are distributed over those n service centers. The $n \times m$ clustering matrix \mathbf{X}_2 specifies which user belongs to which center. Thus

$$x_{2ij} = \begin{cases} 1, & \text{if user } j \text{ belongs to center } i \\ 0, & \text{otherwise.} \end{cases} \quad (33.120)$$

A user can only belong to one center; thus, $\forall j = 1, 2, \dots, m, \sum_{i=1}^n x_{2ij} = 1$. We define an $n \times (n+m)$ matrix \mathbf{X} called the spanning tree matrix $(\mathbf{X}_1 \ \mathbf{X}_2)$. The bicriteria LAN topology design problem can be formulated as the following nonlinear 0-1 programming model [23, 64], where $R(\mathbf{X})$

is the network reliability and w_{1ij} is the weight of the link between the centers i and j :

$$\max R(\mathbf{x}) \tag{33.121}$$

$$\min \sum_{i=1}^{n-1} \sum_{j=i+1}^n w_{1ij}x_{1ij} + \sum_{i=1}^n \sum_{j=1}^m w_{2ij}x_{2ij} \tag{33.122}$$

$$\text{s. t. } \sum_{j=1}^m x_{1ij} \leq g_i, \quad i = 1, 2, \dots, m \tag{33.123}$$

$$\sum_{i=1}^n x_{2ij} = 1, \quad j = 1, 2, \dots, m, \tag{33.124}$$

w_{2ij} is the weight of the link between the center i and the user j , and g_i is the maximum number that can connect to the center i .

Genetic Algorithm Approach

Representation: We can easily construct an encoding as follows:

Procedure: Encoding of Prüfer Number

- Step 1 Let node i be the smallest labeled leaf node in a labeled tree T
- Step 2 Let j be the first digit in the encoding as the node j incident to node i is uniquely determined. The encoding is built by appending digits from left to right.
- Step 3 Remove node i and the link from i to j ; thus we have a tree with $k - 1$ nodes.
- Step 4 Repeat the above steps until one link is left. We produce a Prüfer number or an encoding with $k - 2$ digits between 1 and k inclusive.

Crossover: Uniform crossover is used. This type of crossover is accomplished by selecting two parent solutions and randomly taking a component from one parent to form the corresponding component of the offspring (Fig. 33.34).

Mutation: Swap mutation is used, as explained in the previous section.

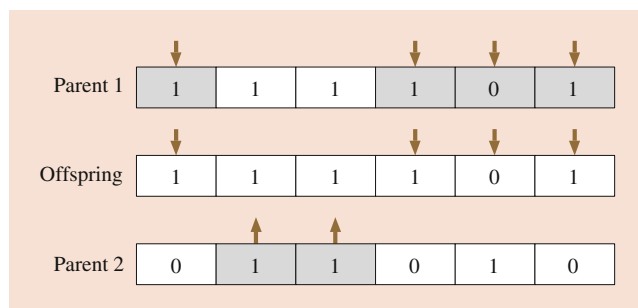


Fig. 33.34 Uniform crossover operator

33.6 Logistics Network Problems

The transportation problem is a basic model in the logistics networks. Many scholars have since refined and extended the basic transportation model to include not only the determination of optimum transportation patterns but also the analysis of production scheduling problems, transshipment problems, and assignment problems.

33.6.1 Linear Transportation Model

The linear transportation problem (LTP) involves the shipment of some homogeneous commodity from various origins or sources of supply to a set of destinations, each demanding specified levels of the commodity. The usual objective function is to minimize the total transportation cost or total weighted distance or to maximize the total profit contribution from the allocation. Given m origins and n destinations, the transportation problem can be formulated as a linear programming model[4, 5]:

$$\min z \sum_{i=1}^m \sum_{j=1}^n c_{ij}x_{ij}, \tag{33.125}$$

$$\text{s. t. } \sum_{j=1}^n x_{ij} \leq a_i, \quad i = 1, 2, \dots, m, \tag{33.126}$$

$$\sum_{i=1}^m x_{ij} \geq b_j, \quad j = 1, 2, \dots, n, \tag{33.127}$$

$$x_{ij} \geq 0, \quad \text{for all } i \text{ and } j \tag{33.128}$$

where x_{ij} is the amount of units shipped from origin i to destination j , c_{ij} is the cost of shipping one unit from source i to destination j , a_i is the number of units available at origin i , and b_j is the number of units demanded at destination j .

Genetic Algorithm Approach

Representation. Perhaps the matrix is the most natural representation of a solution for the transportation problem. The allocation matrix for the transportation problem can be written as follows:

$$X_p = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{nm} \end{pmatrix} \tag{33.129}$$

where X_p denotes the p -th chromosome and x_{ij} is the corresponding decision variable.

Crossover Assume that two matrices $X_1 = (x_{ij}^1)$ and $X_2 = (x_{ij}^2)$ are selected as parents for the crossover operation. The crossover is performed in the following three main steps:

- Step 1 Create two temporary matrices $D = (d_{ij})$ and $R = (r_{ij})$ as follows: $d_{ij} = [(x_{ij}^1 + x_{ij}^2)/2]$ and $r_{ij} = [(x_{ij}^1 + x_{ij}^2)/2]$ mode 2.
- Step 2 Divide matrix R into two matrices $R^1 = (r_{ij}^1)$ and $R^2 = (r_{ij}^2)$ such that $R = R^1 + R^2$
- Step 3 Then we produce two offspring of X_1' and X_2' as follows: $X_1' = D + R^1$ and $X_2' = D + R^2$

Mutation The mutation is performed in following three main steps:

- Step 1 Make a submatrix from the parent matrix. Randomly select i_1, \dots, i_p rows and j_1, \dots, j_q columns to create a $(p * q)$ submatrix $Y = (y_{ij})$, where i_1, \dots, i_p is a proper subset of $1, 2, \dots, m$, i_1, \dots, i_p and $2 \leq p \leq m$, j_1, \dots, j_q is a proper subset of $1, 2, \dots, n$ and $2 \leq q \leq n$, and y_{ij} takes the value of the element in the crossing position of selected row i and column j in the parent matrix.
- Step 2 Reallocate commodity for the submatrix. The available amount of commodity a_i^y and the demands b_j^y for the submatrix are determined as follows:

$$a_i^y = \sum_{j \in \{j_1, \dots, j_q\}} y_{ij}, i = i_1, i_2, \dots, i_p, \quad (33.130)$$

$$b_j^y = \sum_{i \in \{i_1, \dots, i_p\}} y_{ij}, j = j_1, j_2, \dots, j_q, \quad (33.131)$$

- Step 3 Replace appropriate element of the parent matrix by new elements from the reallocated submatrix Y .

Spanning Tree-Based Approach Transportation problems (TP) as a special type of network problem have a special data structure characterized as a transportation graph in their solutions. The spanning tree-based GA incorporating this data structure of TP was proposed by Gen and Li. This GA utilized the Prüfer number encoding based on a spanning tree, which is adopted because it is capable of representing all possible trees. Using the Prüfer number representation, the memory only requires $m + n - 2$ entries for a chromosome in the TP. Transportation problems have separable sets of nodes for plants and warehouses. From this point, Gen and Cheng designed the criterion for feasibility of the chromosome. The proposed spanning tree-based GA can find the optimal or near-optimal solution for transportation problems in the solution space [10, 61, 64].

33.6.2 Multiobjective Transportation

In the transportation problem, multiple objectives are required in practical situations, such as minimizing transportation cost, minimizing the average shipping time to priority customers, maximizing production using a given process, minimizing fuel consumption, and so on. The traditional multiobjective transportation problem (moTP) with m plants and n warehouses can be formulated as

$$\min z_q = \sum_{i=1}^m \sum_{j=1}^n c_{ij}^q x_{ij}, \quad q = 1, 2, \dots, Q, \quad (33.132)$$

$$\text{s. t. } \sum_{j=1}^n x_{ij} \leq a_i, \quad i = 1, 2, \dots, m, \quad (33.133)$$

$$\sum_{i=1}^m x_{ij} \geq b_j, \quad j = 1, 2, \dots, n, \quad (33.134)$$

$$x_{ij} \geq 0, \quad \forall i, j, \quad (33.135)$$

where q means the q -th objective function.

Spanning Tree-Based GA for Multiobjective TP

The Pareto optimal solutions are usually characterized as the solutions of the multiobjective programming problem [37, 65].

33.6.3 Bicriteria Transportation Model with Fuzzy Coefficients

Consider the following two objectives: minimizing total transportation cost and minimizing total delivery time. Let \tilde{c}_{ij}^1 be the fuzzy data representing the transportation cost of shipping one unit from plant i to warehouse j , \tilde{c}_{ij}^2 be the fuzzy data representing the delivery time of shipping one unit of the product from plant i to warehouse j , a_i be the number of units available at plant i , and b_j be the number of units demanded at warehouse j . This problem with m plants and n warehouses can be formulated as

$$\min \tilde{z}_1 = \sum_{i=1}^m \sum_{j=1}^n \tilde{c}_{ij}^1 x_{ij}, \quad (33.136)$$

$$\min \tilde{z}_2 = \sum_{i=1}^m \sum_{j=1}^n \tilde{c}_{ij}^2 x_{ij}, \quad (33.137)$$

$$\text{s. t. } \sum_{j=1}^n x_{ij} \leq a_i, \quad i = 1, 2, \dots, m, \quad (33.138)$$

$$\sum_{j=1}^n x_{ij} \geq b_j, \quad j = 1, 2, \dots, n, \quad (33.139)$$

$$x_{ij} \geq 0, \quad \forall i, j, \quad (33.140)$$

where P and W are upper limits on total number of plants or DCs that can be opened. where x_{ij} is the unknown quantity to be transported from plant i to warehouse j .

Genetic Algorithm Approach

The proposed genetic algorithm approach is based on spanning tree. In multicriteria optimization, we are interested in finding Pareto solutions. When the coefficients of objectives are represented with fuzzy numbers, the objective values become fuzzy numbers. Since a fuzzy number represents many possible real numbers, it is not easy to compare solutions to determine which is the Pareto solution. Fuzzy ranking techniques can help us to compare fuzzy numbers. In this approach, Pareto solutions are determined based on the ranked values of fuzzy objective functions, and genetic algorithms are used to search for Pareto solutions.

Representation The spanning tree encoding, the Prüfer number, is used to represent the candidate solution. The criterion for the solution’s feasibility designed in the proposed spanning tree-based GA is also employed.

Crossover For simplicity one-point crossover is used.

Mutation Inversion mutation and displacement mutation are used.

33.6.4 Multistage Logistics Model

Logistics network design is one of the most important fields of *supply chain management* (SCM). SCM describes the discipline of optimizing the delivery of goods, services, and information from supplier to customer. Typical SCM goals include transportation network design, plant/DC location, production schedule streamlining, and efforts to improve order response time. Logistics network design is one of the most important fields of SCM. It offers great potential to reduce costs and to improve service quality. There are now expanded multistage logistics models for applications to real-world cases so the supply chain network (SCN) design problem is referred to as three-stage, i.e., suppliers, plants, DCs, and customers [65–69]. The design task involves the decision of choosing to open a facility (plant or DCs) or not and the distribution network design to satisfy the customer demand at minimum cost. The three-stage logistics network model is formulated as a 0-1 mixed-integer programming as follows:

$$\begin{aligned} \min & \sum_i \sum_j s_{ij}x_{ij} + \sum_j \sum_k f_{jk}y_{jk} \\ & + \sum_k \sum_l u_{kl}z_{kl} + \sum_j f_j w_j + \sum_k g_k z_k \end{aligned} \quad (33.141)$$

$$\text{s. t. } \sum_j x_{ij} \leq a_i, \forall i \quad (33.142)$$

$$\sum_k y_{jk} \leq b_j w_j, \forall j \quad (33.143)$$

$$\sum_j w_j \leq P \quad (33.144)$$

$$\sum_l z_{kl} \leq c_k z_k, \forall k \quad (33.145)$$

$$\sum_k z_k \leq W, \quad (33.146)$$

$$\sum_k z_{kl} \geq d_l, \forall l \quad (33.147)$$

$$w_j, z_k = (0, 1), \forall j, k, \quad (33.148)$$

$$x_{ij}, y_{jk}, z_{kl} \geq 0, \forall i, j, k, l, \quad (33.149)$$

where P and W are upper limits on total number of plants or DCs that can be opened.

Transportation Tree Representation

For a two-stage transportation problem (tsTP) as a simple example of multistage logistics model, a chromosome consists of priorities for sources and depots to obtain a transportation tree. Its length is equal to the total number of sources ($|K|$) and depots ($|J|$), i.e., $|K| + |J|$.

For a given chromosome, the transportation tree is generated by a sequential arc appending procedure between sources and depots. At each step, only one arc is added to the tree by selecting a source (depot) with the highest priority and connecting it to a depot (source) with the minimum cost. Figure 33.35 represents a transportation tree with three sources and four depots, its cost matrix, and the corresponding priority-based encoding.

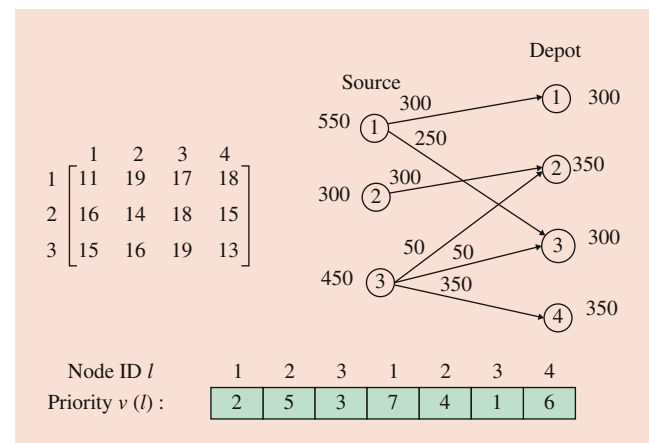


Fig. 33.35 A sample of transportation tree and its encoding

Table 33.8 Trace table of decoding procedure

Iteration	$v(k+j)$	a	b	k	j	g_{kj}
0	[2 5 3 7 4 1 6]	(550, 300, 450)	(300, 350, 300, 350)	1	1	300
1	[2 5 3 0 4 1 6]	(250, 300, 450)	(0, 350, 300, 350)	3	4	350
2	[2 5 3 0 4 1 0]	(250, 300, 100)	(0, 350, 300, 0)	2	2	300
3	[2 0 3 0 4 1 0]	(250, 0, 100)	(0, 50, 300, 0)	3	2	50
4	[2 0 3 0 0 1 0]	(250, 0, 50)	(0, 0, 300, 0)	3	3	50
5	[2 0 0 0 0 1 0]	(250, 0, 0)	(0, 0, 250, 0)	1	3	250
6	[0 0 0 0 0 0 0]	(0, 0, 0)	(0, 0, 0, 0)			

The decoding process by the priority-based encoding is given in Table 33.8. As shown in the trace table, at the first step of the decoding procedure, an arc between Depot 1 and Source 1 is added to the transportation tree since Depot 1 has the highest priority in the chromosome and the lowest cost between Source 1 and Depot 1. After determining the amount of shipment that is $g_{11} = \min\{550, 300\} = 300$, capacity of source and demand of depot are updated as $a_1 = 550 - 300 = 250$ and $b_1 = 300 - 300 = 0$, respectively. Since $b_1 = 0$, the priority of Depot 1 is set to 0, and Depot 4 with the next highest priority is selected.

After adding arc between Depot 4 and Source 3, the amount of shipment between them is determined and their capacity and demand are updated as explained above, and this process repeats until demands of all depots are met.

For the priority-based encoding and decoding algorithms for any transportation tree, we can refer to Section 3.4 Multi-stage Logistics Models in [4].

As shown in Fig. 33.36, we consider a simple example that has three feasible plants, four feasible DCs, and five customers.

As shown in Fig. 33.36, we consider a simple example that has three feasible plants, four feasible DCs, and five customers. When the upper limit of opened DCs is taken as three, the first part of chromosome consists of seven digits, and the second part consists of nine digits. The total capacity of DCs which will be opened has to satisfy the total demand of customers. Considering this property, we first decode the second part of chromosome. In this phase, transportation tree on the second stage and decision related with which DCs will be opened are obtained simultaneously. After that the transportation tree on the first stage is obtained by considering DCs which were opened in the second stage of the decoding procedure. In the second stage, it is possible to obtain a transportation tree that doesn't satisfy the upper limit of opened DCs, since connection between source and depot is realized considering minimum cost.

Genetic Operation

In this multistage logistics model, we adopt the weight mapping crossover (WMX). Similar to crossover, mutation is done to prevent the premature convergence, and it explores new solution space. However, unlike crossover, mutation is usually done by modifying a gene within a chromosome. We also investigated the effects of two different mutation operators on the performance of GA. Insert and swap mutations were used for this purpose. For the selection, we adopt the *roulette wheel selection* (RWS). This is to determine selection probability or survival probability for each chromosome proportional to the fitness value. For the numerical experiments of the various sizes of test problems, refer to Section 3.4 Multi-stage Logistics Models in [4].

33.6.5 SCM Network Design

Supply chain management (SCM) is to choose the subset of plants and distribution centers to be opened and to design the distribution network strategy that can satisfy all capacities and demand requirements imposed by customers with minimum cost. We formulate the problem by using the following mixed-integer linear programming (MILP) model:

$$\min \sum_i \sum_j s_{ij}x_{ij} + \sum_j \sum_k t_{jk}y_{jk} + \sum_k \sum_l u_{kl}x_{kl} + \sum_j f_j x_j + \sum_k g_k x_k \quad (33.150)$$

$$\text{s. t. } \sum_j x_{ij} \leq a_i, \quad \forall i \quad (33.151)$$

$$\sum_k y_{jk} \leq b_j w_j, \quad \forall j \quad (33.152)$$

$$\sum_j w_j \leq P, \quad (33.153)$$

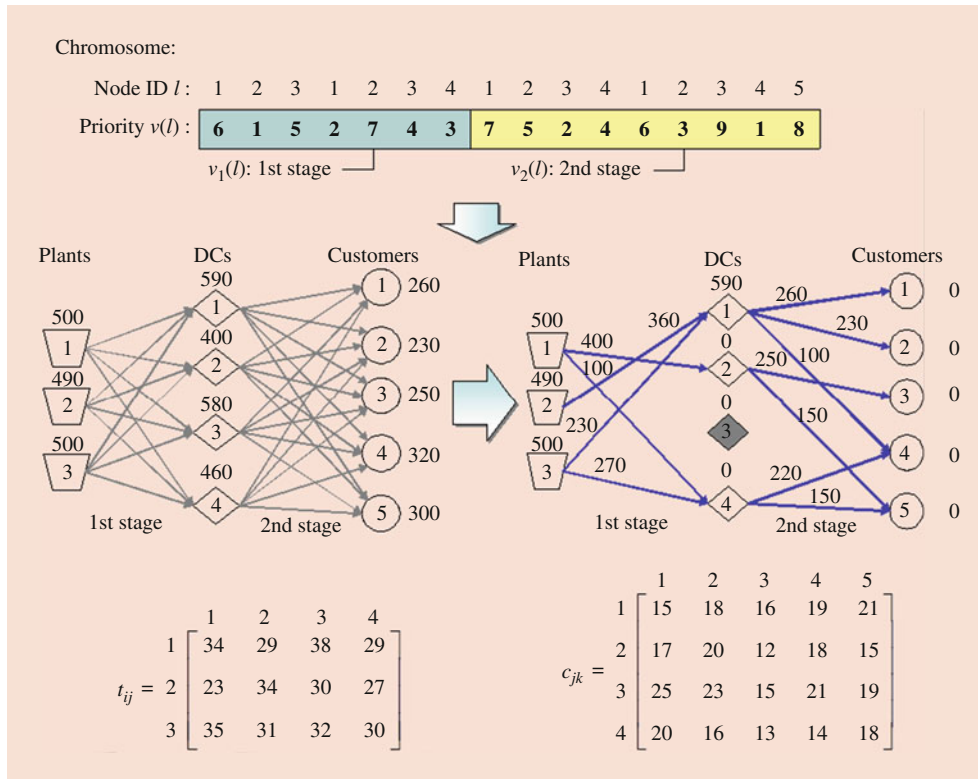


Fig. 33.36 An illustration of chromosome, transportation trees, and transportation costs for each stage on tsTP

$$\sum_l z_{kl} \leq c_k u_k, \forall k \quad (33.154)$$

$$\sum_k u_k \leq W, \quad (33.155)$$

$$\sum_k z_{kl} \geq d_l, \forall l \quad (33.156)$$

$$w_j, u_k = \{0, 1\}, \forall j, k \quad (33.157)$$

$$x_{ij}, y_{jk}, z_{kl} \geq 0, \forall i, j, k, l \quad (33.158)$$

distribution center k , and z_{kl} is the amount shipped from distribution center k to customer l . w_j and u_k are defined as

$$w_j = \begin{cases} 1, & \text{if production takes place at plant } j \\ 0, & \text{otherwise.} \end{cases} \quad (33.159)$$

$$u_k = \begin{cases} 1, & \text{if distribution center } k \text{ is opened} \\ 0, & \text{otherwise.} \end{cases} \quad (33.160)$$

where I is the number of suppliers, J is the number of plants, K is the number of distribution centers, L is the number of customers, a_i is the capacity of supplier i , b_j is the capacity of plant j , c_k is the capacity of distribution center k , d_l is the demand of customer l , s_{ij} is the unit cost of production in plant j using material from supplier i , t_{jk} is the unit cost of transportation from plant j to the distribution center k , u_{kl} is the unit cost of transportation from distribution k to customer l , f_j is the fixed cost for operating plant j , g_k is the fixed cost for operating distribution center k , W is an upper limit on the total number of distribution centers that can be opened, and P is an upper limit on the total number of plants that can be opened.

Here, x_{ij} is the quantity produced at plant j using raw material from supplier i , y_{jk} is the amount shipped from plant j to

Crossover: The crossover is done by exchanging the information of two parents to provide a powerful exploration capability. We employ a one-cut-point crossover operation, which randomly selects one cut-point and exchanges the right parts of the two parents to generate offspring.

Mutation: Modifying one or more of the gene values of an existing individual, mutation creates a new individual to increase the variability of the population. We use inversion and displacement mutation operations.

33.6.6 Reverse Logistics Model

Beyond the current interest in supply chain management, recent attention has been given to extending the traditional forward supply chain to incorporate a reverse logistic element

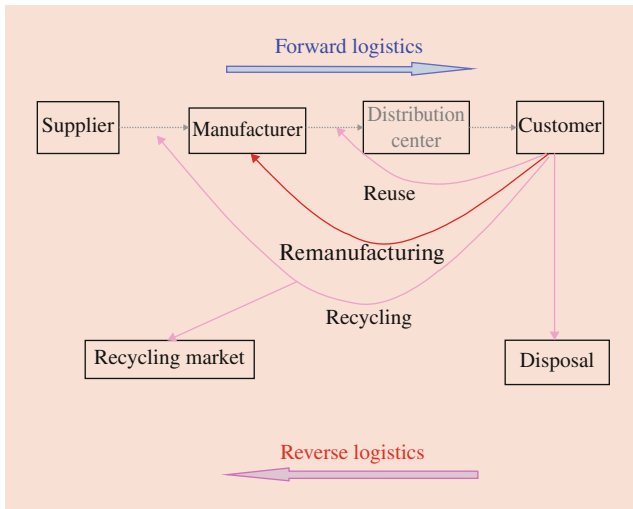


Fig. 33.37 Logistics flow of forward and reverse

owing to liberalized return policies, environmental concern, and a growing emphasis on customer service and parts reuse. Implementation of reverse logistics especially in product returns would allow not only for savings in inventory carrying cost, transportation cost, and waste disposal cost due to returned products but also for the improvement of customer loyalty and future sales.

As shown in Fig. 33.37, the reverse logistics problem (or closed-loop recycling problem) can be formulated as a multistage logistics network model.

First, recovered products from customers are transported to disassembly center, in the first stage. Then disassembled parts are transported to processing center, in the second stage. Last, in the third stage, processed parts are transported to manufacturer. In the third stage, if the quantity of provided parts from processing center is not enough for requirement of manufacturer, then manufacturer must buy parts from suppliers. In case of opposition, then exceeded capacities are distributed in order of recycling and disposal. The illustrative of the multistage reverse logistics network model is shown in Fig. 33.38. For a mathematical model of the multistage reverse logistics network formulated as a 0-1 mixed programming and genetic algorithm based on the priority encoding method, refer to [4, 70]. Recently reported is the multistage reverse logistics network design for product resale by adaptive genetic algorithm [35, 66–69].

Lee et al. [71] proposed a multiobjective hybrid genetic algorithm for minimizing the total cost and delivery tardiness in a reverse logistics designing problem. Guo et al. [72, 73] reported the case study of lead battery in Shanghai for the dynamic joint construction and optimal strategy of multiobjective multi-period multistage government-enterprise reverse logistics network.

33.7 Location and Allocation Problems

Location-allocation problems arise in many practical settings. The classical single location-allocation problem is to find the single location which minimizes the summed distance from some number of fixed points, representing customers with known locations.

33.7.1 Location-Allocation Model

There are m facilities to be located, and n customers with known locations are to be allocated to the variable facilities. Each customer has the requirement $q_j, j = 1, 2, \dots, n$, and each facility has the capacity $b_i, i = 1, 2, \dots, m$. We need to find the locations of facilities and allocations of customers to facilities so that the total summed distance among the customers and their serving facilities is minimized (Fig. 33.39). This problem is formulated mathematically as [24, 74]

$$\min \sum_{i=1}^m \sum_{j=1}^n \sqrt{(x_i - u_j)^2 + (y_i - v_j)^2} z_{ij} \quad (33.161)$$

$$\text{s. t. } \sum_{j=1}^n q_j \cdot z_{ij} \leq b_i, \quad i = 1, 2, \dots, m, \quad (33.162)$$

$$\sum_{i=1}^m z_{ij} = 1, \quad j = 1, 2, \dots, n, \quad (33.163)$$

$$z_{ij} = 0 \text{ or } 1, \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n, \quad (33.164)$$

where

$$(u_j, v_j) = \text{location of customer } j, \quad j = 1, 2, \dots, n, \quad (33.165)$$

$$(x_i, y_i) = \text{location of facility } i,$$

decision variables $i = 1, 2, \dots, m$

$z_{ij} = 0 - 1$ decision variable,

$$z_{ij} = \begin{cases} 1, & \text{customer } j \text{ is served by facility } i \\ 0, & \text{otherwise.} \end{cases} \quad (33.166)$$

A. Genetic Algorithm Approach

Representation: Since location variables are continuous, the float-value chromosome representation is used. A chromosome is given as follows:

$$c^k = [(x_1^k, y_1^k)(x_2^k, y_2^k) \dots (x_m^k, y_m^k)]$$

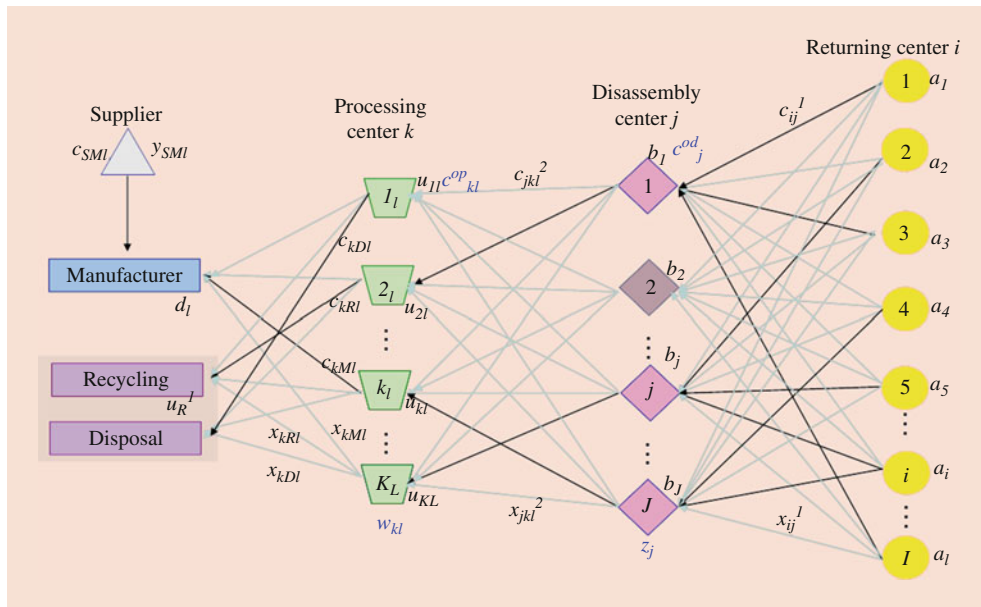


Fig. 33.38 Multistage reverse logistics network model

where (x_i^k, y_i^k) is the location of the i -th facility in the k -th chromosome, $i = 1, 2, \dots, m$.

Crossover: Two mating strategies are used: one is free mating, which selects two parents at random; another is dominating mating, which uses the fittest individual as a fixed parent and randomly selects another parent from the population pool. These two strategies are used alternatively in the evolutionary process. Suppose two parents with the following chromosomes are selected to produce a child:

$$c^{k1} = [(x_1^{k1}, y_1^{k1})(x_2^{k1}, y_2^{k1}) \dots (x_m^{k1}, y_m^{k1})]$$

$$c^{k2} = [(x_1^{k2}, y_1^{k2})(x_2^{k2}, y_2^{k2}) \dots (x_m^{k2}, y_m^{k2})]$$

Only one child is allowed to be produced:

$$c = [(x_1, y_1)(x_2, y_2) \dots (x_m, y_m)]$$

$$x_i = r_i \cdot x_i^{k1} + (1 - r_i)x_i^{k2},$$

$$y_i = r_i \cdot y_i^{k1} + (1 - r_i)y_i^{k2}, \quad (33.167)$$

Mutation: Suppose the candidate chromosome to be mutated is as follows:

$$c^k = [(x_1^k, y_1^k)(x_2^k, y_2^k) \dots (x_m^k, y_m^k)]$$

then the chromosome of the child produced by subtle mutation $c = [x_1, y_1, x_2, y_2, \dots, x_m, y_m]$ is as follows:

$$x_i = x_i^k + \text{random value in } [-\varepsilon, \varepsilon],$$

$$y_i = y_i^k + \text{random value in } [-\varepsilon, \varepsilon], \quad (33.168)$$

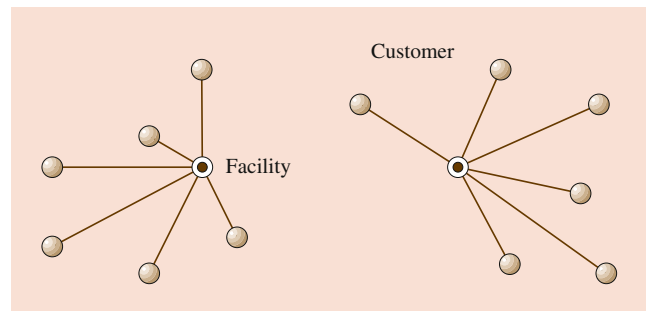


Fig. 33.39 Location-allocation problem

B. Numerical Example

Cooper and Rosing’s examples are used to test the effectiveness of this method. Cooper carefully constructed the front half data which contains three natural groups, and Rosing increased the number of customers with random points. These examples provide a good benchmark to test the effectiveness of the proposed method because their global optimal solutions have already been found.

These examples include 30 customers whose location coordinates are shown in Table 33.9. There is a common location-allocation problem where the requirements of the customers are treated as equal and the capacities of the facilities are assumed to be unlimited.

33.7.2 Capacitated Plant Location Model

The capacitated plant location problem (cPLP) is referred to as a fixed-charge problem to determine the locations of plants

Table 33.9 Coordinates of Cooper and Rosing’s example

Order number	X	Y	Order number	X	Y
1	5	9	16	53	8
2	5	24	17	1	34
3	5	48	18	33	8
4	13	4	19	3	26
5	12	19	20	17	9
6	13	39	21	53	20
7	28	37	22	24	17
8	21	45	23	40	22
9	25	50	24	22	41
10	31	9	25	7	13
11	39	2	26	5	17
12	39	16	27	39	3
13	45	22	28	50	50
14	41	30	29	16	40
15	49	31	30	22	45

with minimal total cost, including production, shipping costs, and fixed costs where the plants are located. In this case, m sources (or facility locations) produce a single commodity for n customers, each with demand of $b_j (j = 1, \dots, n)$ units. If a particular source i is opened (or facility is built), it has a fixed cost $d_i \geq 0$ and a production capacity $a_i \geq 0$ associated with it. There is also a positive cost c_{ij} for shipping a unit from source i to customer j . The problem is to determine the locations of the plants so that capacities are not exceeded and demands are met, all at a minimal total cost. The cPLP is a mixed-integer program, as shown in the following:

$$\min z(x) = \sum_{i=1}^m \sum_{j=1}^n c_{ij}x_{ij} + \sum_{i=1}^m d_i y_i \quad (33.169)$$

$$\text{s. t. } \sum_{i=1}^m x_{ij} = b_j, \quad j = 1, 2, \dots, n \quad (33.170)$$

$$\sum_{j=1}^n x_{ij} \leq a_i y_i, \quad i = 1, 2, \dots, m \quad (33.171)$$

$$x_{ij} \geq 0, \quad \forall i, j \quad (33.172)$$

$$y_i = 0 \text{ or } 1, \quad i = 1, 2, \dots, m, \quad (33.173)$$

The variables are x_{ij} and y_i , which represent the amount shipped from plant i to warehouse j and whether a plant is open (or located) ($y_i = 1$) or closed ($y_i = 0$), respectively.

Spanning Tree-Based GA for Plant Location Problems

The spanning tree-based GA for the capacitated plant location problem is the same as that of the fixed-charge

transportation problem except there is a different evaluation function in the evolutionary process[61].

33.7.3 Obstacle Location-Allocation Model

There are n customers with known locations and m facilities to be built to supply some kind of services to all customers, for example, supplying materials or energy. There are also p obstacles representing some forbidden areas. The formulation of the mathematical model is based on the following assumptions:

- Customer j has service demand $q_j, j = 1, 2, \dots, n$,
- Facility i has service capacity $b_i, i = 1, 2, \dots, m$,
- Each customer should be served by only one facility,
- New facilities should not be built within any obstacle,
- Connecting paths between facilities and customers should not be allowed to pass through any of the obstacles.

The problem is to choose the best locations for facilities so that the sum of distances between customers and their serving facilities is minimal, as illustrated in Fig. 33.40. The obstacle location-allocation problem can be formulated as follows:

$$\min f(D, z) = \sum_{i=1}^m \sum_{j=1}^n t(D_i, C_j) \cdot z_{ij} \quad (33.174)$$

$$\text{s. t. } \sum_{j=1}^m z_{ij} = 1, \quad i = 1, 2, \dots, m \quad (33.175)$$

$$\sum_{i=1}^m d_j z_{ij} = b_j, \quad j = 1, 2, \dots, n \quad (33.176)$$

$$D_i = (x_i, y_i) \notin Q_k, \quad i = 1, 2, \dots, m, \quad (33.177)$$

$$k = 1, 2, \dots, q,$$

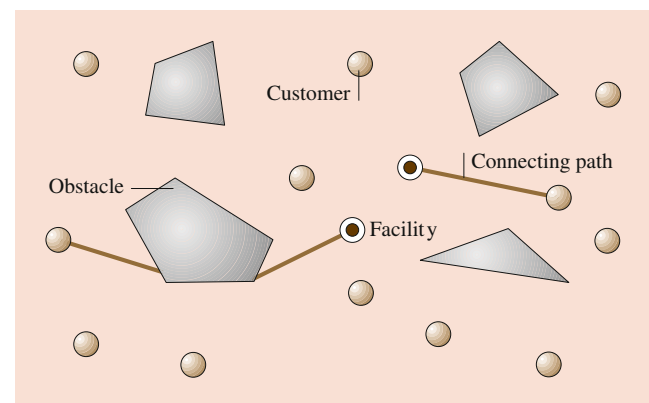


Fig. 33.40 Obstacle location-allocation problem

$$(x_i, y_i) \in R_T, i = 1, 2, \dots, m, \tag{33.178}$$

$$x_i, y_i \in R_T, i = 1, 2, \dots, m, \tag{33.179}$$

$$z_{ij} = 1 \text{ or } 0, i = 1, 2, \dots, m, j = 1, 2, \dots, n, \tag{33.180}$$

where $C_j = (u_j, v_j)$ is the location of the j -th customer, $D_i = (x_i, y_i)$ is the decision variable, the location of the j -th distribution center DC_i should not fall within any of the obstacles, $t(D_i, C_j)$ is the shortest connecting path from the set of possible paths between the distribution center DC_i and the customer C_j which avoids all obstacles, R_T is the total area considered for the location and allocation problem, and z_{ij} is a 0–1 decision variable; $z_{ij} = 1$ indicates that the j -th customer is served by DC_i , $z_{ij} = 0$ otherwise.

Hybrid Evolutionary Method

Since there are obstacles, the locations of the chromosome produced by initialization, crossover, and mutation procedure may become infeasible. Generally, there are three kinds of methods to treat infeasible chromosomes as shown in Table 33.10. The first is to discard it, but according to the experience of other researchers, this method may lead to very low efficiency. The second is to add a penalty to infeasible chromosomes. The third is to repair the infeasible chromosome according to the characteristics of the specified problem.

33.7.4 LAP with Multi-facility Service

LAP Model with Multiple facility Service The sugarcane loading with multi-facility problem is identified as a location-allocation problem (LAP) for multi-facility services in sugarcane supply system. The m loading stations are considered as candidate facilities. They are selected to satisfy n sugarcane

fields (C_i) each of which contains a_i tons of sugarcane, $i = 1, 2, \dots, n$. We assume that the capacity of a facility is specified or known. The several k types of facility services (L_{jk}) are assigned for each loading station (F_j). This model is formulated as a 0–1 mixed-integer programming (0-1 MIP) model for locating a number of loading stations among a finite set of potential sites and assigning the several types of facility services to each loading station and allocating sugarcane between these loading stations and growers’ fields.

Based on the characteristics of the LAP, the mathematical model will be constructed as follows: The objective function is to minimize the total cost in any period (week or month) within 1 crop year in which the first term is the transportation cost from a field to a loading station. The second term is the transportation cost from the loading station to a sugar mill. The third term is the fixed cost of each loading station. Finally, the fourth and fifth terms are the operation cost of loading sugarcane to a truck or trailer and the fixed cost of each transloader, respectively:

$$f(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{i=1}^n \sum_{j=1}^m h d_{ij} x_{ij} + \sum_{j=1}^m h l_{jM} y_j + \sum_{j=1}^m e_j y_j + \sum_{j=1}^m \sum_{k=1}^p o_k z_{jk} + \sum_{j=1}^m \sum_{k=1}^p f_k z_{jk} \tag{33.181}$$

Seven constraints are formulated as follows:

$$g_j(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^n a_i x_{ij} \leq B_j y_j, j = 1, 2, \dots, m \tag{33.182}$$

$$g_{m+i}(\mathbf{x}) = \sum_{j=1}^m x_{ij} = 1, j = 1, 2, \dots, m \tag{33.183}$$

Table 33.10 Comparison results of Cooper and Rosing’s example

Problem n/m	Rosing’s method optimal objective	ALA		HEM	
		Best	Percent error	Best	Percent error
15/2	214.281	219.2595	2.32	214.2843	0.0015
15/3	143.197	144.8724	1.17	143.2058	0.0061
15/4	113.568	115.4588	1.69	113.5887	0.0182
15/5	97.289	99.4237	2.19	97.5656	0.2843
15/6	81.264	84.0772	3.46	83.0065	2.14
30/2	447.728	450.3931	0.5952	447.73	0.0004
30/3	307.372	310.3160	0.9578	307.3743	0.0007
30/4	254.148	258.4713	1.7010	254.2246	0.0301
30/5	220.057	226.8971	3.1083	220.4335	0.1711
30/6	–	208.4301	3.4940	201.4031	0.0

F_1			F_2			F_3			F_4			F_5					
1			0			0			1			1					
14	12	13	10	7	9				5	4	1	8	11	6	15	3	2
1	1	1							1	1	0	2	0	0			
L_1	L_2	L_3	L_1	L_2	L_3	L_1	L_2	L_3	L_1	L_2	L_3	L_1	L_2	L_3			

Fig. 33.41 Illustration of a representation chromosome

$$g_{m+n+1}(\mathbf{y}) = \sum_{j=1}^m y_j \leq M \tag{33.184}$$

$$g_{m+n+1+j}(\mathbf{z}) = \sum_{i=1}^n a_i x_{ij} \leq \sum_{k=1}^p b_k z_{jk}, \quad j = 1, 2, \dots, m \tag{33.185}$$

$$x_{ij} = 0 \text{ or } 1, \quad i = 1, 2, \dots, n \quad i = 1, 2, \dots, m \tag{33.186}$$

$$y_j = 0 \text{ or } 1, \quad j = 1, 2, \dots, n \tag{33.187}$$

$$z_{jk} \geq 0 \text{ Integer}, \quad j = 1, 2, \dots, m, \quad k = 1, 2, \dots, p \tag{33.188}$$

Constraints (33.182) ensure that the service capacity of each loading station shall not exceed its capacity. Constraints (33.183) ensure that every field shall be served by only one loading station. Constraints (33.183) reflect that total number of opened loading stations shall not exceed the total number of candidate loading stations. Constraints (33.185) ensure that the total quantity of sugarcane can be served by each loading station. Constraints (33.186–33.188) are variable restrictions for the decision variables to be found.

Designing Representation and Feasibility of Chromosome For designing a natural representation of LAP for multi-facility services, an effective chromosome representation was built by employing three parts of chromosomes in the encoding method. The design of the chromosome and its parameters is by necessity specific to the problem to be solved, and each chromosome in the population consists of three parts as shown in the example in Fig. 33.41:

- (1) Facility location (or loading station’s location; $F_j, j = 1, 2, \dots, m$): 0–1 chromosome.
- (2) Customer allocation (or sugarcane allocation; $C_i, i = 1, 2, \dots, n$): Heuristics-based chromosome.
- (3) Facility service assignment (or transloader assignment; $L_k, k = 1, 2, \dots, p$): Integer-based chromosome.

The procedure of initialization based on heuristics is given as follows:

- Step 1 Calculate the distance between each sugarcane field and each loading station.
- Step 2 Find the minimum distance between each sugarcane field and loading stations.
- Step 3 Assign all sugarcane fields to the nearest loading stations with adequate available capacity. If not, then the sugarcane field will be assigned to the loading station which is the second nearest to the field and so on.

Genetic Operations

Crossover Generally, crossover generates offspring that combines both parents’ features by exchanging the information of each. Based on the characteristics of the chromosome, we apply displacement crossover.

Check-and-Repair Problem Constraints for Offspring

Although GAs are powerful and popular optimization tools because of their simplicity and ease of implementation, they do not incorporate constraint-handling features. Hence, they may encounter difficulties when applied for solving highly constrained problems. This is because the search operators of GAs (i.e., crossover and mutation) are blind to constraints [74] which can result in infeasible offspring. Therefore, a proper constraint-handling method is required to maintain the applicability of GAs with the solutions satisfying the constraints.

Mutation This operator produces random changes in a chromosome of a population generation. By using swapping mutation operators with one or more genes are swapped by randomly selecting two genes within a chromosome and exchanging their positions with one another.

For designing a hybrid genetic algorithm, fuzzy logic controller (FLC) is combined to automatically tune crossover and mutation parameters by Yun and Gen [7, 8] who reported a case study of a sugar industry in northeastern Thailand in which there are 14 sugar mills out of a total 47 sugar mills in the entire country. The case study’s sugar mill produces

approximately 23,000 tons of sugarcane per day over an area of 4.68×10^4 [ha] in eight provinces, and for the detailed contents, refer to [74].

Acknowledgments This work is supported by the Grant-in-Aid for Scientific Research (C) of the Japan Society for the Promotion of Science (JSPS: 19K12148).

References

- Goldberg, D.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Reading (1989)
- Gen, M., Lin, L.: Genetic algorithms. In: Wah, B. (ed.) Wiley Encyclopedia of Computer Science and Engineering. John Wiley & Sons, Hoboken (2008)
- Yu, X., Gen, M.: Introduction to Evolutionary Algorithms. Springer, London (2010)
- Gen, M., Cheng, R., Lin, L.: Network Models and Optimization: Multiple Objective Genetic Algorithm Approach. Springer, London (2008)
- Michalewicz, Z.: Genetic Algorithm + Data Structures = Evolution Programs. Springer, New York (1994)
- Gen, M., Cheng, R.: Evolutionary network design: hybrid genetic algorithms approach. *Int. J. Comput. Intell. Appl.* **3**, 357–380 (2003)
- Yun, Y., Gen, M.: Adaptive hybrid genetic algorithm with fuzzy logic controller. In: Verdegay, J.L. (ed.) Fuzzy Sets Based Heuristics for Optimization, pp. 251–263. Springer, Berlin (2003)
- Yun, Y., Gen, M.: Performance analysis of adapted genetic algorithm with fuzzy logic and heuristics. *Fuzzy Optim. Decis. Making* **2**, 161–175 (2003)
- Lin, L., M, Gen: auto-tuning strategy for evolutionary algorithms: balancing between exploration and exploitation. *Soft Comp.* **13**(2), 157–168 (2009)
- Gen, M., Cheng, R.: Genetic Algorithms & Engineering Optimization. Wiley, New York (2000)
- Yun, Y.S., Gen, M., Hwang, R.K.: Adaptive genetic algorithm to multi-stage reverse logistics network design for product resale. *Information* **15**(12), 6117–6138 (2013)
- Deb, K.: Multiobjective Optimization Using Evolutionary Algorithms. Wiley, Chichester (2001)
- Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: improving the strength Pareto evolutionary algorithm. *TIK-report* **103**, 1–21 (2001)
- Lin, L., Gen, M.: Multiobjective genetic algorithm for bicriteria network design problems. In: Intelligent and Evolutionary Systems, SCI 187, pp. 141–161. Springer, Berlin (2009)
- Zhang, W.Q., Gen, M., Jo, J.B.: Hybrid sampling strategy-based multiobjective evolutionary algorithm for process planning and scheduling problem. *J. Intell. Manuf.* **25**(5), 881–897 (2014)
- Zhang, W.Q., Yang, D., Zhang, G., Gen, M.: Hybrid multiobjective evolutionary algorithm with fast sampling strategy-based global search and route sequence difference-based local search for VRPTW. *Expert Syst. App.* **145**, 1–16 (2020)
- Goldberg, D.E., Holland, J.H.: Genetic algorithms and machine learning. *Mach. Learn.* **3**(2), 95–99 (1988)
- Lin, L., Gen, M.: Hybrid evolutionary optimization with learning for production scheduling: state-of-the-art survey on algorithms and applications. *Int. J. Prod. Res.* **56**(1–2), 193–223 (2018)
- Jourdan, L., Dhaenens, C., Talbi, E.G.: Using datamining techniques to help metaheuristics: a short survey. *Hybrid. Metahe.* **4030**, 57–69 (2006)
- Zhang, J., Zhan, Z., Lin, Y., Chen, N., Gong, Y., Zhong, J., Chung, H., Li, Y., Shi, Y.: Evolutionary computation meets machine learning: a survey. *IEEE Comput. Intell. Mag.* **6**(4), 68–75 (2011)
- Martello, S., Toth, P.: Knapsack Problems: Algorithms and Computer Implementations. Wiley, Chichester (1990)
- Lin, L., Gen, M.: Node-based genetic algorithm for communication spanning tree problem. *IEICE Trans. Comm.* **E89-B**(4), 1091–1098 (2006)
- Zhou, G., Gen, M.: A genetic algorithm approach on tree-like telecommunication network design problem. *J. Oper. Res. Soc.* **54**, 248–254 (2003)
- Davis, L. (ed.): Handbook of Genetic Algorithms. Van Nostrand Reinhold, New York (1991)
- Gen, M., Cheng, R.: Genetic Algorithms & Engineering Design. Wiley, New York (1997)
- Goldberg, D., Lingle, R.: Loci and the traveling salesman problem. In: Proceedings of the First International Conference on Genetic Algorithms, New Jersey, pp. 154–192 (1985)
- Cheng, R., Gen, M.: Evolution program for resource constrained project scheduling problem. In: Orlando, D.F. (ed.) Proceedings of First IEEE Conference on Evolutionary Computation, pp. 736–741 (IEEE, Piscataway, 1994)
- Gen, M., Kumar, A., Kim, J.R.: Recent network design techniques using evolutionary algorithms. *Int. J. Prod. Econ.* **98**(2), 251–261 (2005)
- Cheng, R., Gen, M.: Resource constrained project scheduling problem using genetic algorithm. *Inter. J. Intell. Autom. Soft Comp.* **3**, 273–286 (1997)
- Yokota, T., Gen, M., Ida, K., Taguchi, T.: Optimal design of system reliability by an approved genetic algorithm. *Elect. Commun. Jpn.* **79**(2), 41–51 (1996)
- Hao, X.C., Gen, M.: Multiplicative job shop rescheduling by using evolutionary algorithm. *IEEJ Trans. Electron. Inf. Syst.* **131**(5), 674–681 (2011)
- Gen, M., Lin, L., Cheng, R.: Bicriteria network optimization problem using priority-based genetic algorithm. *IEEJ Trans. Elect. Inf. Syst.* **124**, 1972–1978 (2004)
- Gen, M., Lin, L.: Multi-objective hybrid genetic algorithm for bicriteria network design problem. *Complex. Int.* **11**, 73–83 (2005)
- Gen, M., Lin, L., Jo, J.B.: Evolutionary network design by multiobjective hybrid genetic algorithm. In: Gen, M., et al. (eds.) Intelligent and Evolutionary Systems, SCI 187, pp. 105–121. Springer, Berlin (2009)
- Baker, K.: Introduction to Sequencing and Scheduling. Wiley, New York (1974)
- Gen, M., Tsujimura, Y., Kubota, E.: Solving job-shop scheduling problem using genetic algorithms. In: Proceedings of IEEE International Conference on Systems Man & Cybernetics, pp. 1577–1582 (1994)
- Gen, M., Lin, L., Yun, Y.S., Inoue, H.: Recent advances in hybrid priority-based genetic algorithms for logistics and SCM network design. *Comput. Ind. Eng.* **115**, 394–412 (2018)
- Gen, M., Lin, L.: Multiobjective evolutionary algorithm for manufacturing scheduling problems: state-of-the-art survey. *J. Intell. Manuf.* **25**(5), 849–866 (2014)
- Cheng, R., Gen, M., Tsujimura, Y.: A tutorial survey of job-shop scheduling problems using genetic algorithms, part I: representation. *Comput. Ind. Eng.* **30**(4), 983–997 (1996)
- Cheng, R., Gen, M., Tsujimura, Y.: A tutorial survey of job-shop scheduling problems using genetic algorithms, part II: hybrid genetic search strategies. *Comput. Ind. Eng.* **36**(2), 343–364 (1999)

41. Kacem, I., Hammadi, S., Borne, P.: Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems. *IEEE Trans. Syst. Man Cybern. Part C* **32**(1), 1–13 (2002)
42. Zhang, H., Gen, M.: Multistage-based genetic algorithm for flexible job-shop scheduling problem. *J. Complex. Inter.* **11**, 223–232 (2005)
43. Sun, L., Lin, L., Gen, M., Li, H.: A hybrid cooperative co-evolution algorithm for fuzzy flexible job shop scheduling. *IEEE Trans. Fuzzy Syst.* **27**(5), 1008–1022 (2019)
44. Gao, J., Gen, M., Sun, L.: Scheduling jobs and maintenances in flexible job shop with a hybrid genetic algorithm. *J. Intell. Manuf.* **17**(4), 493–508 (2006)
45. Gao, J., Gen, M., Sun, L., Zhao, X.: A hybrid of genetic algorithm and bottleneck shifting for multiobjective flexible job shop scheduling problems. *Comp. Ind. Eng.* **53**(1), 149–162 (2007)
46. Gao, J., Sun, L., Gen, M.: A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems. *Comp. Oper. Res.* **35**(9), 2892–2907 (2008)
47. Gen, M., Gao, J., Lin, L.: Multistage-based genetic algorithm for flexible job-shop scheduling problem. In: *Intelligent and Evolutionary Systems*, pp. 183–196. Springer, Berlin (2009)
48. Gen, M., Lin, L., Zhang, H.: Evolutionary techniques for optimization problems in integrated manufacturing system: state-of-the-art survey. *Comp. Ind. Eng.* **56**(3), 779–808 (2009)
49. Chou, C.-W., Chien, C.-F., Gen, M.: A multiobjective hybrid genetic algorithm for TFT-LCD module assembly scheduling. *IEEE Trans. Automat. Sci. Eng.* **11**(3), 692–705 (2014)
50. Jamrus, T., Chien, C.-F., Gen, M., Sethanan, K.: Hybrid particle swarm optimization combined with genetic operators for flexible job-shop scheduling under uncertain processing time for semiconductor manufacturing. *IEEE Trans. Semicon. Manuf.* **31**(1), 32–41 (2018)
51. Sun, L., Lin, L., Li, H., Gen, M.: Large scale flexible scheduling optimization by a distributed evolutionary algorithm. *Comp. Ind. Eng.* **128**, 894–904 (2019)
52. Tian, J., Hao, X.C., Gen, M.: A hybrid multi-objective EDA for robust resource constraint project scheduling with uncertainty. *Comput. Ind. Eng.* **128**, 317–326 (2019)
53. Yoo, M., Gen, M.: Scheduling algorithm for real-time tasks using multiobjective hybrid genetic algorithm in heterogeneous multiprocessors system. *Comput. Oper. Res.* **34**(10), 3084–3098 (2007)
54. Tsujimura, Y., Gen, M., Kubota, E.: Solving fuzzy assembly line balancing using genetic algorithms. *Comp. Ind. Eng.* **29**(1–4), 543–547 (1995)
55. Scholl, A.: *Balancing and Sequencing of Assembly Lines*. Physica-Verlag, Heidelberg (1999)
56. Zhang, W., Xu, W.T., Liu, G., Gen, M.: An effective hybrid evolutionary algorithm for stochastic multiobjective assembly line balancing problem. *J. Intell. Manuf.* **28**, 783–790 (2017)
57. Lin, L., Gen, M., Gao, J.: Optimization and improvement in robot-based assembly line system by hybrid genetic algorithm. *IEEJ Trans. Electr. Inform. Syst.* **128**(3), 424–431 (2008)
58. Gao, J., Sun, L., Wang, L., Gen, M.: An efficient approach for type II robotic assembly line balancing problems. *Comp. Indus. Eng.* **56**(3), 1065–1080 (2009)
59. Zhang, W., Gen, M.: An efficient multiobjective genetic algorithm for mixed-model assembly line balancing problem considering demand ratio-based cycle time. *J. Intell. Manuf.* **22**(3), 367–378 (2011)
60. Zhang, W., Xu, W.T., Liu, G., Gen, M.: An effective hybrid evolutionary algorithm for stochastic multiobjective assembly line balancing problem. *J. Intell. Manuf.* **28**, 783–790 (2017)
61. Jo, J.B., Li, Y.Z., Gen, M.: Nonlinear fixed charge transportation problem by spanning tree-based genetic algorithm. *Comput. Ind. Eng.* **52**, 290–298 (2007)
62. Lee, C.Y., Yun, Y., Gen, M.: Reliability optimization design for complex systems by hybrid GA with fuzzy logic controller and local search. *IEICE Trans. Electron.* **E85-A**, 880–891 (2002)
63. Lin, L., Gen, M.: A self-control genetic algorithm for reliable communication network design. In: *Proceedings of IEEE Congress on Evolutionary Computation Vancouver*. IEEE Press, Piscataway (2006)
64. Syarif, A., Gen, M.: Solving exclusionary side constrained transportation problem by using a hybrid spanning tree-based genetic algorithm. *J. Intell. Manuf.* **14**, 389–399 (2003)
65. Gen, M., Syarif, A.: Hybrid genetic algorithm for multi-time period production/distribution planning. *Comput. Ind. Eng.* **48**(4), 799–809 (2005)
66. Syarif, A., Yun, Y.S., Gen, M.: Study on multi-stage logistics chain network: a spanning tree-based genetic algorithm approach. *Comput. Ind. Eng.* **43**, 299–314 (2002)
67. Gen, M., Syarif, A.: Multi-stage supply chain network by hybrid genetic algorithms with fuzzy logic controller. In: *Verdegay, J.L. (ed.) Fuzzy Sets Based Heuristics for Optimization*, pp. 181–196. Springer, Berlin (2003)
68. Zhou, G., Min, H., Gen, M.: A genetic algorithm approach to the bi-criteria allocation of customers to warehouses. *Int. J. Prod. Econ.* **86**, 35–45 (2003)
69. Gen, M., Altiparmak, F., Lin, L.: A genetic algorithm for two-stage transportation problem using priority-based encoding. *OR Spectr.* **28**, 337–354 (2006)
70. Lee, J., Gen, M., Rhee, K.: Network model and optimization of reverse logistics by hybrid genetic algorithm. *Comp. Indus. Eng.* **56**(3), 951–964 (2009)
71. Lee, J., Chung, K., Lee, K., Gen, M.: A multi-objective hybrid genetic algorithm to minimize the total cost and delivery tardiness in a reverse logistics. *Multimed. Tools Appl.* **74**(20), 9067–9085 (2015)
72. Guo, J.Q., Wang, X.Y., Fan, S.Y., Gen, M.: Dynamic joint construction and optimal strategy of multi-objective multi-period multi-stage government-enterprise reverse logistics network: A case study of lead battery in Shanghai. *Comp. Indus. Eng.* **106**, 351–360 (2017)
73. Guo, J.Q., He, L., Gen, M.: Optimal strategies for the closed-loop supply chain with the consideration of supply disruption and subsidy policy. *Comp. Indus. Eng.* **128**, 886–893 (2019)
74. Neungnatcha, W., Sethanan, K., Gen, M., Theerakulpisut, S.: Adaptive genetic algorithm for solving sugarcane loading stations with multi-facility services problem. *Comput. Electron. Agric.* **98**, 85–99 (2013)
75. Gen, M., Yun, Y.: Soft computing approach for reliability optimization: state-of-the-art survey. *Reliab. Eng. Syst. Saf.* **91**, 1008–1026 (2006)



Mitsuo Gen received his PhD from Kogakuin Univ. in Tokyo in 1974. He was faculties at Ashikaga Institute of Tech. for 1974–2003 and Waseda Univ. for 2003–2010. He was visiting faculties at Univ. of California, Berkeley for 1999–2000, Texas A&M Univ. for 2000, Hanyang Univ. for 2010–2012, and National Tsing Hua Univ. for 2012–2014. He published several books on Genetic Algorithms including “Network Models and Optimization” from Springer, London in 2008.



Lin Lin received the Ph.D. degree from Waseda University, Japan, in 2008. He is a Professor with the International School of Information Science and Engineering, Dalian University of Technology, China, and a Senior Researcher with Fuzzy Logic Systems Institute, Japan. His research interest includes computational intelligence and their applications in combinatorial optimization and pattern recognition.



Deterministic and Stochastic DCA for DC Programming

34

Hoai An Le Thi , Tao Pham Dinh , Hoang Phuc Hau Luu, and Hoai Minh Le

Contents

34.1	Introduction	675
34.2	Deterministic DC Programming and DCA	676
34.2.1	Preliminaries.....	677
34.2.2	DC Programming and DCA: Basic Results.....	678
34.2.3	Links Between DCA and Standard Convex/Nonconvex Programming Approaches.....	682
34.2.4	Proximal Upper Bounds of Convex Functions.....	683
34.3	Stochastic Optimization in the Literature	685
34.4	Stochastic DC Programming and DCA	687
34.4.1	A Class of Stochastic DC Programs.....	687
34.4.2	Online Stochastic DCA1.....	688
34.4.3	Online Stochastic DCA2.....	692
34.5	Conclusion	696
34.6	Appendix A: A Theorem on Conditional Expectation	696
34.7	Appendix B: Uniform Strong Law of Large Number	697
B.1	Uniform Strong Law of Large Number for a Sequence.....	698
B.2	Uniform Strong Law of Large Number for Triangular Arrays.....	698
References	699

Some of these problems admit nonconvex objective functions. On the other hand, DCA (difference of convex functions algorithm) has proven its strength in tackling a large class of smooth or nonsmooth, nonconvex optimization problems called DC programming. The key advantages of DCA come from its simplicity and flexibility that allows it to treat large-scale problems arising in various contexts. This chapter concerns methods incorporating ideas of stochastic optimization in an online manner into DCA framework to create new algorithms called online stochastic DCA. The first section introduces the chapter. The second section accounts for deterministic DC programming and DCA. The third section briefly reviews stochastic optimization. The fourth section is dedicated to stochastic DC programming and DCA, where we propose two online stochastic DCA schemes for solving a class of stochastic DC programs. The last section concludes the chapter with discussions about promising aspects of the topic.

Keywords

DC programming · DCA · Stochastic DC programming · Online stochastic DCA · Conditional expectation

Abstract

In the context of big data analysis, stochastic optimization algorithms are widely used as effective tools to handle data complexity and data uncertainty. These algorithms usually aim to solve problems modeled as stochastic programs.

H. A. Le Thi (✉) · H. P. H. Luu · H. M. Le
University of Lorraine, Metz, France

Institut Universitaire de France (IUF), Paris, France
e-mail: hoai-an.le-thi@univ-lorraine.fr;
hoang-phuc-hau.luu@univ-lorraine.fr; minh.le@univ-lorraine.fr

T. Pham Dinh
National Institute of Applied Sciences – Rouen, Rouen, France
e-mail: pham-dinh.tao@insa-rouen.fr

34.1 Introduction

In many real-life problems, uncertainty appears naturally due to the insufficiency of our knowledge. The unknown quantity in which we are interested is modeled as a random variable or a random vector called Z . Suppose that we have to make a decision called w prior to knowing Z . When a realization z of Z becomes known, depending on the decision which was made, we have to pay a cost of $f(w, z)$. The optimization problem involved consists of finding the optimal decision w minimizing the cost $f(w, z)$. Since the realization z is unknown before the decision is made, one reasonable approach is minimizing the cost on average, i.e., the expected cost (the

expected loss) considered is $F(\mathbf{w}) = E(f(\mathbf{w}, \mathbf{Z}))$. Furthermore, if we have a piece of information \mathcal{P} about \mathbf{Z} , the loss should be updated as $F_{\mathcal{P}}(\mathbf{w}) = E(f(\mathbf{w}, \mathbf{Z})|\mathcal{P})$. Obviously, if \mathcal{P} is independent to \mathbf{Z} , its information is useless, and we then recover $E(f(\mathbf{w}, \mathbf{Z})|\mathcal{P}) = E(f(\mathbf{w}, \mathbf{Z}))$. In contrast, when \mathcal{P} gives enough information to determine \mathbf{Z} , a realization z of \mathbf{Z} becomes known, and the following relation holds:

$$E(f(\mathbf{w}, \mathbf{Z})|\mathcal{P}) = f(\mathbf{w}, z).$$

Solving stochastic programming problems is a challenging task because the underlying distribution of \mathbf{Z} is usually unknown. If sampling independently from the distribution of \mathbf{Z} is possible, one can solve the empirical loss instead. However, the empirical loss is well-approximated to the expected loss only if the sample size is large enough. Hence, solving directly the empirical loss could be very expensive. The main idea of stochastic optimization in an online manner consists of using one or several samples at a time, combining with information of the previous iteration to create a new optimization problem. The new problem is simpler than the original problem with the empirical loss, so it is more likely easier to solve. On top of that, stochastic optimization is the most applicable regarding sequential data since it gives elegant ways to integrate new information into the previous solution.

In order to model accurately real-world problems, stochastic programming framework usually takes into account nonconvexity. When it relates to nonconvexity, things become complicated and they require novel robust tools. However, even in deterministic nonconvex programming, finding effective algorithms that ensure both the quality of solutions and the scalability for large-scale problems is a great challenge. In this deterministic context, it is well-known that most nonconvex nonsmooth problems encountered in real life fall into the framework of DC programming. In nonconvex programming, DCA is one of the rare algorithms that allow to solve problems in a large-scale setting. This renowned method is used by researchers and practitioners worldwide and is usually proven to be state-of-the-art comparing with other related standard methods [1–3].

Despite of its efficiency and scalability in deterministic nonconvex optimization, traditional DCA finds it difficult to apply directly to solve stochastic programming problems due to the lack of knowledge of the distribution. For this reason, it is natural to develop stochastic versions of DCA.

In this chapter, we propose two stochastic DCA schemes for solving a class of stochastic DC problems. Our main idea is quite simple; we replace some deterministic quantities in the DCA by their corresponding stochastic approximations. The stochastic approximations are constructed in an online manner since we are interested in sequential data coming from the distribution of \mathbf{Z} . Our algorithms, on the one hand,

confront nonconvexity. On the other hand, they handle the stochastic nature of the problems.

34.2 Deterministic DC Programming and DCA

It goes, without saying, that in applied mathematics and computer science, nonconvex (differentiable or nondifferentiable) programming have experienced over the past two decades dramatic developments in the world. The explanation for this explosion is quite simple: on the one hand, the variational optimization approaches are flexible and efficient alongside the classical ones, which know their limits on qualities of numerical solutions and sizes of tackled problems; and on the other hand, modern convex analysis and convex optimization widely studied since the early 1960s (at the time being, one can say that available theoretical and algorithmic tools might allow to formulate/reformulate suitable convex programs and to design efficient related solution algorithms) are forced to a logical and natural extension to nonconvexity/nondifferentiability. As most real-world optimization problems are nonconvex and in many areas of industrial, economic, and financial applications, current requirements lead to replacements of older convex models by nonconvex ones, more complex (especially for large-scale problems) but more reliable (because they represent more accurately the nature of considered problems), and especially more economical and competitive.

The absence of convexity is a source of difficulties of all kinds, namely, the distinction between the local and global minima and the nonexistence of verifiable characterizations of global solutions, etc. Passing from convex to nonconvex programming causes all the computational complexity. If one word is used to differentiate between easy and hard problems, convexity is probably the watershed (dixit Rockafellar). Finding a global solution of nonconvex programs is the holy grail of the optimization community.

Really realistic and pragmatic extension of convex analysis, convex programming to nonconvex analysis, nonconvex programming, should be sufficiently large to cover most real-world nonconvex programs, but not too much broad in order to use the powerful arsenal of convex analysis/programming.

We hear more and more often speaking of *trading convexity for scalability* and *the blessings and the curses of dimensionality* in areas such as data mining and machine learning, transport and logistic, communication networks, computational biology, finance and economy, image science, etc., to name but a few, and those now grouped into the discipline called data science, where we have to deal with very large nonconvex programs. Consequently, one should rather look for inexpensive, scalable, and efficient local approaches for the large-scale setting.

Idea (philosophy/principle):

If there is a problem you can't solve, then there is an easier problem you can solve: find it. (George Polya)

1. How to approximate a DC function by a convex function to get a convex program.
2. What to do with a solution \mathbf{x}^* of such a convex program.
3. How to iterate the process.

Reply to these three questions implies the construction of DCA which consists in iteratively generating a sequence $\{\mathbf{x}^k\}$ of solutions of successive convex subprograms. For standard DC programs (i.e., with only convex constraints), one can state local/global optimality conditions by using just basic modern convex analysis (convex analysis by Rockafellar, 1970 [4]), more precisely fundamental properties of conjugate and subdifferential of convex functions.

DC programming and DCA, which constitute the backbone of nonconvex programming, are introduced in 1985 by Pham Dinh Tao in the preliminary state, as a quite natural and logical extension of his previous works on concave programming at the early 1970s, and extensively developed by Le Thi Hoai An and Pham Dinh Tao since 1994 to become now classic and increasingly popular. Being a descent method without line search, DCA, which is one of rare effective algorithms in nonsmooth nonconvex programming, has proven its strength in this framework, especially in a large-scale setting. Regardless of its local behavior, DCA finds itself quite often reaching a global solution if the initial point is good enough. The DC structures of the objective and constraint functions are exploited by DCA in a suitable way which allows us to make use of powerful tools in convex optimization. In fact, solving a DC program using DCA amounts to solving successively convex optimization problems. That is, at each iteration, the method minimizes a convex surrogate of the objective. By its way of incorporating convex optimization in each iteration, DCA can be considered as a natural extension of convex optimization to DC programming. Moreover, with suitable DC decomposition of the objective, DCA can recover most of existing algorithms in nonconvex programming.

With the main strength coming from its deep mathematical foundation, simplicity and flexibility, DCA has found its applications in numerous areas of applied sciences. We list here some areas using DCA (with just a few citations): data mining and machine learning [5–7], finance [8, 9], transport logistics [10, 11], image processing [12, 13], computer vision [14], cryptology [15], computational chemistry and biology [16, 17], game theory [18], telecommunications [19], multiobjective programming [20], inverse problems and ill-posed problems [21], mathematical programming with equilibrium constraints [22, 23], etc. The reader can see the list of reference in [1, 2].

34.2.1 Preliminaries

This subsection recalls some basic notions from convex analysis and nonsmooth analysis [4, 24, 25] to make preparations for giving the theory of DC programming and DCA.

The space $X := \mathbb{R}^n$ is equipped with the canonical inner product $\langle \cdot, \cdot \rangle$. Its dual space Y is identified with X itself. The effective domain of a function $f : X \rightarrow \mathbb{R} \cup \{+\infty\}$, denoted by $\text{dom} f$, is

$$\text{dom} f := \{\mathbf{x} \in X : f(\mathbf{x}) < +\infty\}.$$

It is called proper if $\text{dom} f \neq \emptyset$. Let C be a nonempty and convex set contained in $\text{dom} f$. By definition, f is convex on C if the following inequality holds for all $\mathbf{x}, \mathbf{u} \in C, \lambda \in [0, 1]$:

$$f(\lambda \mathbf{x} + (1 - \lambda)\mathbf{u}) \leq \lambda f(\mathbf{x}) + (1 - \lambda)f(\mathbf{u}).$$

Let $\rho \geq 0$; f is called ρ -convex on C if $f - \frac{\rho}{2}\|\cdot\|^2$ is convex on C . The modulus of strong convexity of f on C , denoted by $\rho(f, C)$ or $\rho(f)$ if $C = \text{dom} f$, is defined by

$$\rho(f, C) := \sup\{\rho \geq 0 : f - (\rho/2)\|\cdot\|^2 \text{ is convex on } C\}.$$

Obviously, ρ -convexity implies convexity. The function f is called strongly convex on C if $\rho(f, C) > 0$. Let $\mathbf{x} \in \text{dom} f$; a vector $\mathbf{z} \in X$ is called a subgradient of f at \mathbf{x} if

$$\forall \mathbf{y} \in X, f(\mathbf{y}) - f(\mathbf{x}) \geq \langle \mathbf{z}, \mathbf{y} - \mathbf{x} \rangle.$$

The whole set of subgradients of f at \mathbf{x} , denoted by $\partial f(\mathbf{x})$, is called the subdifferential of f at \mathbf{x} . In general, $\partial f(\mathbf{x})$ can be empty. We say that f is subdifferentiable at \mathbf{x} if $\partial f(\mathbf{x}) \neq \emptyset$.

The set $A \subset X$ is called an affine space if

$$\forall t \in \mathbb{R}, \forall \mathbf{x}, \mathbf{y} \in A : \mathbf{x} + t(\mathbf{y} - \mathbf{x}) \in A.$$

Let $C \subset X$; the affine hull of C , denoted by $\text{aff}(C)$, is the smallest affine space containing C .

The relative interior of a convex set C , denoted by $\text{ri} C$, is defined as follows:

$$\text{ri} C := \{\mathbf{x} \in \text{aff}(C) \mid \exists \epsilon > 0, B(\mathbf{x}, \epsilon) \cap \text{aff}(C) \subset C\},$$

where $B(\mathbf{x}, \epsilon)$ is the open ball with the center \mathbf{x} and the radius ϵ .

We next recall two concepts of subdifferential, namely, Fréchet subdifferential and Clarke subdifferential.

Let $f : X \rightarrow \mathbb{R} \cup \{+\infty\}$ be a proper function; the Fréchet subdifferential of f at $\mathbf{x} \in \text{dom} f$, denoted by $\partial^F f(\mathbf{x})$, is defined as follows:

$$\partial^F f(\mathbf{x}) = \left\{ \mathbf{y} \in Y : \liminf_{\mathbf{h} \rightarrow 0} \frac{f(\mathbf{x} + \mathbf{h}) - f(\mathbf{x}) - \langle \mathbf{y}, \mathbf{h} \rangle}{\|\mathbf{h}\|} \geq 0 \right\}. \quad (P_{dc}) \quad \alpha := \inf \{f(\mathbf{x}) := g(\mathbf{x}) - h(\mathbf{x}) : \mathbf{x} \in X\}$$

For $\mathbf{x} \notin \text{dom } f$, we set $\partial^F f(\mathbf{x}) = \emptyset$. If f is a convex function, then $\partial^F f$ coincides with the subdifferential ∂f . A point $\mathbf{x}_0 \in X$ is called a (Fréchet) *critical* (or *stationary*) point for the function f , if $0 \in \partial^F f(\mathbf{x}_0)$.

If f is locally Lipschitz at $\mathbf{x} \in X$, then the Clarke directional derivative $f^C(\mathbf{x}; \cdot)$ at \mathbf{x} along a direction $\mathbf{d} \in X$ is defined by

$$f^C(\mathbf{x}; \mathbf{d}) := \lim_{(t, \mathbf{u}) \rightarrow (0^+, \mathbf{x})} \frac{f(\mathbf{u} + t\mathbf{d}) - f(\mathbf{u})}{t}.$$

The Clarke subdifferential set of f at \mathbf{x} is defined as

$$\partial^C f(\mathbf{x}) := \{\mathbf{y} \in Y : \langle \mathbf{y}, \mathbf{d} \rangle \leq f^C(\mathbf{x}; \mathbf{d}), \forall \mathbf{d} \in X\}.$$

If f is continuously differentiable at \mathbf{x} , $\partial^C f(\mathbf{x})$ coincides with the Fréchet derivative of f at \mathbf{x} , $\{\nabla f(\mathbf{x})\}$. When f is a convex function, then $\partial^C f$ coincides with the subdifferential ∂f . The two basic operators to manipulate the Clarke subdifferential are presented as follows:

$$\partial^C(-f)(\mathbf{x}) = -\partial^C f(\mathbf{x}),$$

$$\partial^C(f + g)(\mathbf{x}) \subset \partial^C f(\mathbf{x}) + \partial^C g(\mathbf{x}),$$

where f, g are locally Lipschitz at a given $\mathbf{x} \in X$. The equality in the latter inclusion holds if f is continuously differentiable at \mathbf{x} . For a DC function f , i.e., $f := g - h$, where g and h are convex functions, one has

$$\partial^F f(\mathbf{x}) \subset \partial g(\mathbf{x}) - \partial h(\mathbf{x}),$$

wherever h is continuous at \mathbf{x} . Especially, if h is differentiable at \mathbf{x} , then the equality holds:

$$\partial^F f(\mathbf{x}) = \partial g(\mathbf{x}) - \nabla h(\mathbf{x}).$$

34.2.2 DC Programming and DCA: Basic Results

Let $\Gamma_0(X)$ denote the convex cone of all lower semicontinuous proper convex functions on X . The vector space of DC functions is denoted by

$$DC(X) := \Gamma_0(X) - \Gamma_0(X).$$

The richness of this class of DC functions allows it to contain most objective functions arising in real-life problems.

The standard DC program takes the form

with $g, h \in \Gamma_0(X)$. Such a function f is called DC on X , $g - h$ is DC decomposition, while g and h are DC components of f . One can, without loss of generality, assume that g and h are strongly convex based on the fact that we can replace g and h by $g + \frac{\rho}{2}\|\cdot\|^2$ and $h + \frac{\rho}{2}\|\cdot\|^2$ with $\rho > 0$. It also implies that a DC function f has infinitely many DC decompositions. Since DCA works on each DC component of f , the DC decomposition highly affects the practical performance of DCA including quality of computed solutions, computational efficiency and scalability, etc.

Note that a DC program with closed convex constraint $\mathbf{x} \in C$ can be equivalently written as a standard DC program by adding the indicator function χ_C to the first component g :

$$\begin{aligned} \inf \{f(\mathbf{x}) := g(\mathbf{x}) - h(\mathbf{x}) : \mathbf{x} \in C\} \\ = \inf \{\chi_C(\mathbf{x}) + g(\mathbf{x}) - h(\mathbf{x}) : \mathbf{x} \in X\}. \end{aligned}$$

Under the convention $+\infty - (+\infty) = +\infty$, the finiteness of the optimal value α implies

$$\text{dom } f = \text{dom}(g + \chi_C) = \text{dom } g \cap \text{dom } \chi_C = C \subset \text{dom } h.$$

The dual DC program (D_{dc}) of (P_{dc}) is

$$(D_{dc}) \quad \alpha = \inf \{h^*(\mathbf{y}) - g^*(\mathbf{y}) : \mathbf{y} \in Y\}.$$

Here g^* and h^* are the conjugate functions of g and h , respectively, i.e.,

$$g^*(\mathbf{y}) := \sup \{\langle \mathbf{x}, \mathbf{y} \rangle - g(\mathbf{x}) : \mathbf{x} \in X\}.$$

Critical and Strongly Critical Points of $g - h$

A point \mathbf{x}^* is a critical point of (P_{dc}) (or $f = g - h$), iff $\partial g(\mathbf{x}^*) \cap \partial h(\mathbf{x}^*) \neq \emptyset$, or equivalently $0 \in \partial g(\mathbf{x}^*) - \partial h(\mathbf{x}^*)$, while it is called strongly critical point of (P_{dc}) (or $f = g - h$), iff $\emptyset \neq \partial h(\mathbf{x}^*) \subset \partial g(\mathbf{x}^*)$. The sets of critical (resp. strongly critical) points of $g - h$ are denoted by \mathcal{P}_c (resp. \mathcal{P}_{sc}), and \mathcal{P}_l is the set of local minimizers of $g - h$. Likewise, replacing (P_{dc}) by (D_{dc}) gives the similar notations for $h^* - g^*$.

Due to the symmetry between the primal DC program and the dual DC program, the following necessary local optimality conditions for the primal problem can be deduced to the dual problem without difficulty [3, 26, 27].

The notion of DC criticality has a close connection to Clarke/Fréchet stationarity that is presented in the following.

The point \mathbf{x}^* is called Clarke/Fréchet stationary if $0 \in \partial^C f(\mathbf{x}^*)$ (resp. $0 \in \partial^F f(\mathbf{x}^*)$). On the other hand, under technical assumptions, the following inclusions hold:

$$\partial^C f(\mathbf{x}) \subset [\partial g(\mathbf{x}) - \partial h(\mathbf{x})],$$

$$\partial^F f(\mathbf{x}) \subset [\partial g(\mathbf{x}) - \partial h(\mathbf{x})].$$

As a consequence, Clarke stationarity of \mathbf{x}^* or its Fréchet stationarity implies *DC criticality* of \mathbf{x}^* . Equivalence between them depends on conditions that make these inclusions become equalities.

DC Strong Criticality and d-Stationarity

As mentioned, DC criticality and strong criticality depend on DC decomposition of DC objective function $f = g - h$. Since a DC function has infinitely many DC decompositions, it is natural and important to find relations between DC criticality and some stationarity concepts that depend only on f . Therefore, we present the commonly used directional stationarity and point out its link with DC criticality.

Some useful results of these different criticalities are presented as follows [4, 28].

Let $\varphi : X \rightarrow \mathbb{R} \cup \{+\infty\}$ be a proper function on X and $\mathbf{x} \in \text{dom } \varphi$. The directional derivative $\varphi'(\mathbf{x}; \cdot)$ of φ at \mathbf{x} along a direction $\mathbf{u} \in X$ is defined by

$$\varphi'(\mathbf{x}; \mathbf{u}) := \lim_{t \downarrow 0} \frac{\varphi(\mathbf{x} + t\mathbf{u}) - \varphi(\mathbf{x})}{t}.$$

If φ is convex, then it becomes

$$\varphi'(\mathbf{x}; \mathbf{u}) := \inf_{t > 0} \frac{\varphi(\mathbf{x} + t\mathbf{u}) - \varphi(\mathbf{x})}{t}.$$

Assume that the function $\varphi : X \rightarrow \mathbb{R} \cup \{+\infty\}$ is proper convex on X ; then for $\mathbf{x} \in \text{dom } \varphi$,

(a) $\varphi'(\mathbf{x}; \cdot)$ is convex positively homogeneous.

(b) $\mathbf{y} \in \partial\varphi(\mathbf{x})$ if and only if $\varphi'(\mathbf{x}; \mathbf{u}) \geq \langle \mathbf{u}, \mathbf{y} \rangle$ for every $\mathbf{u} \in X$, i.e.,

$$\varphi'(\mathbf{x}; \mathbf{u}) \geq \sup\{\langle \mathbf{u}, \mathbf{y} \rangle : \mathbf{y} \in \partial\varphi(\mathbf{x})\} = \chi_{\partial\varphi(\mathbf{x})}^*(\mathbf{u}).$$

In fact, the closure of $\varphi'(\mathbf{x}; \mathbf{u})$ as a convex function of \mathbf{u} is the support function of the closed convex set $\partial\varphi(\mathbf{x})$

$$\text{cl}(\varphi'(\mathbf{x}; \mathbf{u})) = \chi_{\partial\varphi(\mathbf{x})}^*(\mathbf{u}).$$

Moreover, if $\mathbf{x} \in \text{ri}(\text{dom } \varphi)$, then $\partial\varphi(\mathbf{x})$ is nonempty, and $\varphi'(\mathbf{x}; \mathbf{u})$ is semicontinuous and proper on X as a function of \mathbf{u} such that

$$\varphi'(\mathbf{x}; \mathbf{u}) = \sup\{\langle \mathbf{u}, \mathbf{y} \rangle : \mathbf{y} \in \partial\varphi(\mathbf{x})\} = \chi_{\partial\varphi(\mathbf{x})}^*(\mathbf{u}).$$

Let C be a nonempty closed convex set in X . Then, for $\mathbf{x} \in C$,

$$\partial\chi_C(\mathbf{x}) = \{\mathbf{y} \in Y : \langle \mathbf{u} - \mathbf{x}, \mathbf{y} \rangle \leq 0 \forall \mathbf{u} \in C\} = N(C, \mathbf{x}),$$

where $N(C, \mathbf{x})$ is the (closed convex) normal cone to C at \mathbf{x} .

(c) Let C_1 and C_2 be two closed convex sets in X . Then,

$$C_1 \subset C_2 \Leftrightarrow \chi_{C_1}^* \leq \chi_{C_2}^*.$$

The d-Stationarity of a Proper Function

A vector $\mathbf{x} \in \text{dom } \varphi$ is *d-stationary* of φ , iff $\varphi'(\mathbf{x}; \mathbf{u}) \geq 0$ for every $\mathbf{u} \in X$. The following theorem presents the key relations between d-stationarity and strong criticality. The proof follows directly from the mentioned properties a), b), and c).

Theorem 34.1 *Let $f := g - h$ be the DC objective function of the primal DC program (P_{dc}) with $g, h \in \Gamma_0(X)$ and its optimal value α being finite (implying $\text{dom } f = \text{dom } g \subset \text{dom } h$). Then, the following properties hold:*

- (1) $f'(\mathbf{x}; \mathbf{u}) = g'(\mathbf{x}; \mathbf{u}) - h'(\mathbf{x}; \mathbf{u})$ for all $\mathbf{u} \in X$.
- (2) *The vector $\mathbf{x} \in \text{dom } f$ is d-stationary of f (or for (P_{dc})) if and only if $g'(\mathbf{x}; \mathbf{u}) \geq h'(\mathbf{x}; \mathbf{u})$ for all $\mathbf{u} \in X$. Hence for $\mathbf{x} \in \text{dom } f$, if both equalities $g'(\mathbf{x}; \mathbf{u}) = \chi_{\partial g(\mathbf{x})}^*(\mathbf{u})$ and $h'(\mathbf{x}; \mathbf{u}) = \chi_{\partial h(\mathbf{x})}^*(\mathbf{u})$ hold for all $\mathbf{u} \in X$, then there is an identity between the d-stationarity of \mathbf{x} and the strong criticality of \mathbf{x} .*
- (3) *If $\mathbf{x} \in \text{ri}(\text{dom } g) \cap \text{ri}(\text{dom } h)$, then the d-stationarity of \mathbf{x} is equivalent to the strong criticality of \mathbf{x} .*
- (4) *If $\mathbf{x} \in \text{ri}(\text{dom } g)$, then the d-stationarity of \mathbf{x} implies the strong criticality of \mathbf{x} .*
- (5) *If $\mathbf{x} \in \text{ri}(\text{dom } h)$, then the strong criticality of \mathbf{x} implies the d-stationarity of \mathbf{x} .*
- (6) *If the affine hulls of $\text{dom } g$ and of $\text{dom } h$ are identical, then $\text{ri}(\text{dom } g) \subset \text{ri}(\text{dom } h)$, and for $\mathbf{x} \in \text{ri}(\text{dom } g)$, the d-stationarity of \mathbf{x} is equivalent to the strong criticality of \mathbf{x} .*

Duality and Local Optimality Conditions in DC Programming

The following theorem and its corollaries give some important results about DC programming including its duality, (strict) local optimality, and the transportation of local minimizer between primal and dual DC problems [1–3, 26, 27]. These results are useful to understand and analyze DCA.

Theorem 34.2 *Let \mathbf{x}^* be a critical point of $g - h$ and $\mathbf{y}^* \in \partial g(\mathbf{x}^*) \cap \partial h(\mathbf{x}^*)$. Let U be a neighborhood of \mathbf{x}^* such that $(U \cap \text{dom } g) \subset \text{dom } \partial h$. If for any $\mathbf{x} \in U \cap \text{dom } g$ there is $\mathbf{y} \in \partial h(\mathbf{x})$ such that*

$$h^*(\mathbf{y}) - g^*(\mathbf{y}) \geq h^*(\mathbf{y}^*) - g^*(\mathbf{y}^*),$$

then \mathbf{x}^ is a local minimizer of $g - h$. More precisely,*

$$g(\mathbf{x}) - h(\mathbf{x}) \geq g(\mathbf{x}^*) - h(\mathbf{x}^*), \quad \forall \mathbf{x} \in U \cap \text{dom } g.$$

Corollary 34.1 (Sufficient Local Optimality) *Let \mathbf{x}^* be a point that admits a neighborhood U such that $\partial h(\mathbf{x}) \cap \partial g(\mathbf{x}^*) \neq \emptyset, \forall \mathbf{x} \in U \cap \text{dom } g$. Then, \mathbf{x}^* is a local minimizer of $g - h$. More precisely,*

$$g(\mathbf{x}) - h(\mathbf{x}) \geq g(\mathbf{x}^*) - h(\mathbf{x}^*), \quad \forall \mathbf{x} \in U \cap \text{dom } g.$$

Corollary 34.2 (Sufficient Strict Local Optimality) *If $\mathbf{x}^* \in \text{int}(\text{dom } h)$ verifies $\partial h(\mathbf{x}^*) \subset \text{int}(\partial g(\mathbf{x}^*))$, then \mathbf{x}^* is a strict local minimizer of $g - h$.*

Corollary 34.3 (DC Duality Transportation of a Local Minimizer) *Let $\mathbf{x}^* \in \text{dom } \partial h$ be a local minimizer of $g - h$, and let $\mathbf{y}^* \in \partial h(\mathbf{x}^*)$ (i.e., $\partial h(\mathbf{x}^*)$ is nonempty and \mathbf{x}^* admits a neighborhood U such that*

$$g(\mathbf{x}) - h(\mathbf{x}) \geq g(\mathbf{x}^*) - h(\mathbf{x}^*), \quad \forall \mathbf{x} \in U \cap \text{dom } g.$$

If

$$\mathbf{y}^* \in \text{int}(\text{dom } g^*) \text{ and } \partial g^*(\mathbf{y}^*) \subset U \quad (34.1)$$

((34.1) holds if g^* is differentiable at \mathbf{y}^*), then \mathbf{y}^* is a local minimizer of $h^* - g^*$.

DCA and Its Convergence Properties

Philosophy of DCA: As a convex analysis approach to DC programming, the main idea of DCA is quite simple: at the iteration k , the second DC component h is approximated by its affine minorization at \mathbf{x}^k ,

$$h_k(\mathbf{x}) := h(\mathbf{x}^k) + \langle \mathbf{x} - \mathbf{x}^k, \mathbf{y}^k \rangle$$

with $\mathbf{y}^k \in \partial h(\mathbf{x}^k)$. The resulting function is convex and DCA minimizes it to get the next point \mathbf{x}^{k+1} .

Basic DCA scheme.

Initialization: Let $\mathbf{x}^0 \in \text{dom } \partial h$.

For $k = 0, 1, 2 \dots$

Step 1: Compute $\mathbf{y}^k \in \partial h(\mathbf{x}^k)$.

Step 2: Compute

$$\mathbf{x}^{k+1} \in \arg \min \{g(\mathbf{x}) - h_k(\mathbf{x}) : \mathbf{x} \in X\} (P_k).$$

Step 3: If $f(\mathbf{x}^{k+1}) = f(\mathbf{x}^k)$, then stop: \mathbf{x}^k and \mathbf{x}^{k+1} are critical points of (P_{dc}) .

Despite of its simple idea, DCA is a powerful method which enjoys some nice properties mainly coming from local optimality conditions and duality in DC programming.

The following theorem recalls some major results on DCA's convergence [3, 26, 27].

Theorem 34.3 *The DCA is a descent method without line search, but with global convergence, which enjoys the following properties (C and D are two convex sets in X , containing the sequences $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$, respectively):*

- (1) *The sequences $\{g(\mathbf{x}^k) - h(\mathbf{x}^k)\}$ and $\{h^*(\mathbf{y}^k) - g^*(\mathbf{y}^k)\}$ are decreasing and convergent to the same limit, and*
 - *$g(\mathbf{x}^{k+1}) - h(\mathbf{x}^{k+1}) = g(\mathbf{x}^k) - h(\mathbf{x}^k)$ iff $[\rho(g, C) + \rho(h, C)]\|\mathbf{x}^{k+1} - \mathbf{x}^k\| = 0, \mathbf{y}^k \in \partial g(\mathbf{x}^k) \cap \partial h(\mathbf{x}^k)$, and $\mathbf{y}^k \in \partial g(\mathbf{x}^{k+1}) \cap \partial h(\mathbf{x}^{k+1})$. Moreover if g or h is strictly convex on C , then $\mathbf{x}^k = \mathbf{x}^{k+1}$. In such a case, the DCA terminates at the k^{th} iteration (finite convergence of DCA).*
 - *$h^*(\mathbf{y}^{k+1}) - g^*(\mathbf{y}^{k+1}) = h^*(\mathbf{y}^k) - g^*(\mathbf{y}^k)$ iff $\mathbf{x}^{k+1} \in \partial g^*(\mathbf{y}^k) \cap \partial h^*(\mathbf{y}^k), \mathbf{x}^{k+1} \in \partial g^*(\mathbf{y}^{k+1}) \cap \partial h^*(\mathbf{y}^{k+1})$, and $[\rho(g^*, D) + \rho(h^*, D)]\|\mathbf{y}^{k+1} - \mathbf{y}^k\| = 0$. Moreover if g^* or h^* is strictly convex on D , then $\mathbf{y}^{k+1} = \mathbf{y}^k$. In such a case, the DCA terminates at the k^{th} iteration (finite convergence of DCA).*
- (2) *If $\rho(g, C) + \rho(h, C) > 0$ (resp. $\rho(g^*, D) + \rho(h^*, D) > 0$), then the series $\{\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2\}$ and $\{\|\mathbf{y}^{k+1} - \mathbf{y}^k\|^2\}$ are convergent.*
- (3) *If the optimal value α of the problem (P_{dc}) is finite, let \mathbf{x}^* be a limit point of $\{\mathbf{x}^k\}$, then*

$$\frac{\rho(g, C) + \rho(h, C)}{2} \sum_{l=0}^{k-1} \|\mathbf{x}^{l+1} - \mathbf{x}^l\|^2 \leq f(\mathbf{x}^0) - f(\mathbf{x}^*).$$

As a consequence,

$$\begin{aligned} & \frac{k}{2} (\rho(g, C) + \rho(h, C)) \times \min \{ \|\mathbf{x}^l - \mathbf{x}^{l+1}\|^2 : l = \overline{0, k-1} \} \\ & \leq f(\mathbf{x}^0) - f(\mathbf{x}^*). \end{aligned}$$

If $\rho(g, C) + \rho(h, C) > 0$, then

$$\min \{ \|\mathbf{x}^l - \mathbf{x}^{l+1}\| : l = \overline{0, k-1} \} \leq \left(\frac{2(f(\mathbf{x}^0) - f(\mathbf{x}^*))}{k(\rho(g, C) + \rho(h, C))} \right)^{\frac{1}{2}},$$

and consequently, the sequence

$$\{\eta^k\} := \left\{ \min \{ \|\mathbf{x}^l - \mathbf{x}^{l+1}\| : l = \overline{0, k-1} \} \right\}_k$$

converges to 0 with the convergence rate $O(k^{-1/2})$, where k is the number of iterations. This implies that in order to achieve $\eta^k \leq \epsilon$, we must run $O(\epsilon^{-2})$ iterations of DCA.

- (4) *If the optimal value α of the problem (P_{dc}) is finite, let \mathbf{x}^* be a limit point of $\{\mathbf{x}^k\}$, then*

$$\min \{ f(\mathbf{x}^l) - f(\mathbf{x}^{l+1}) : l = \overline{0, k-1} \} \leq \frac{f(\mathbf{x}^0) - f(\mathbf{x}^*)}{k}.$$

Consequently, the sequence

$$\{\mu^k\} := \left\{ \min \{f(\mathbf{x}^l) - f(\mathbf{x}^{l+1}) : l = \overline{0, k-1}\} \right\}_k$$

converges to 0 with the convergence rate $O(k^{-1})$, where k is the number of iterations. This implies that in order to achieve $\mu^k \leq \epsilon$, we must run $O(\epsilon^{-1})$ iterations of DCA.

- (5) If the optimal value α of problem (P_{dc}) is finite and the sequences $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$ are bounded, then every limit point $\tilde{\mathbf{x}}$ (resp. $\tilde{\mathbf{y}}$) of the sequence $\{\mathbf{x}^k\}$ (resp. $\{\mathbf{y}^k\}$) is a critical point of $g - h$ (resp. $h^* - g^*$).
- (6) For polyhedral DC program (i.e., a DC program in which at least one of the functions g and h is polyhedral), the sequences $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$ contain finitely many iterates, and hence they have finite convergence (i.e., after a finite number of iterations).

In general, establishing the convergence rate for the whole sequence $\{\mathbf{x}^k\}$ generated by DCA is a challenging task. Thus, Theorem 34.3(5) only ensures that the sequence $\{\mathbf{x}^k\}$ admits a subsequence converging to the a critical point of $g - h$. It is worth noting that, if $\rho(g, C) + \rho(h, C) > 0$, Theorem 34.3(3) implies that the series

$$\sum_{k=0}^{\infty} \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2$$

is convergent. On the other hand, if the series

$$\sum_{k=0}^{\infty} \|\mathbf{x}^k - \mathbf{x}^{k+1}\| \quad (34.2)$$

is convergent, $\{\mathbf{x}^k\}$ is a Cauchy sequence and, as a consequence, it is convergent. In [29], the authors studied the convergence of (34.2) with the help of Łojasiewicz subgradient inequality for DC programs with subanalytic data. The main results of [29] are summarized as follows with the convergence rates provided.

DCA with Subanalytic Data

Firstly, let us recall some notions of subanalytic sets and functions with their basic properties [30–32].

Definition 34.1

- (1) A subset C of X is said to be semianalytic iff for each point of X , there exists a neighborhood V such that $C \cap V$ is of the following form:

$$C \cap V = \bigcup_{i=1}^p \bigcap_{j=1}^q \{\mathbf{x} \in V : f_{ij}(\mathbf{x}) = 0, g_{ij}(\mathbf{x}) > 0\},$$

where $f_{ij}, g_{ij} : V \rightarrow \mathbb{R}$ ($1 \leq i \leq p, 1 \leq j \leq q$) are real-analytic functions.

- (2) A subset C of X is called subanalytic iff for each point of X , there exists a neighborhood V such that

$$C \cap V = \{\mathbf{x} \in X : \exists \mathbf{y} \in \mathbb{R}^m, (\mathbf{x}, \mathbf{y}) \in D\},$$

where D is a bounded semianalytic subset of $X \times \mathbb{R}^m$ with $m \geq 1$.

- (3) A function $f : X \rightarrow \mathbb{R} \cup \{+\infty\}$ is said to be subanalytic iff its graph $\text{gph} f$ is a subanalytic subset of $X \times \mathbb{R}$.

The class of subanalytic sets (resp. functions) contains all analytic sets (resp. functions). The class of subanalytic sets are closed under locally finite union and intersection, complement and under the natural projection. The distance function to a subanalytic set is a subanalytic function; the sum or difference of the continuous subanalytic functions is also subanalytic. We refer the reader to [31, 33, 34] for further properties.

The subanalyticity of the conjugate of subanalytic function is stated as follows.

Proposition 34.1 *If $f : X \rightarrow \mathbb{R} \cup \{+\infty\}$ is a lower semicontinuous subanalytic strongly convex function, then its conjugate f^* is a $C^{1,1}$ (the class of functions whose derivative is Lipschitz) subanalytic convex function.*

Let us recall next the Łojasiewicz subgradient inequality established by Bolte-Daniliidis-Lewis [35].

Theorem 34.4 (Theorem 3.1 [35]) *Let $f : X \rightarrow \mathbb{R} \cup \{+\infty\}$ be a subanalytic function such that its domain $\text{dom} f$ is closed and $f|_{\text{dom} f}$ is continuous, and let \mathbf{x}_0 be a Fréchet critical point of f . Then, there exist $\theta \in [0, 1)$, $L > 0$ and a neighborhood V of \mathbf{x}_0 such that the following inequality holds:*

$$|f(\mathbf{x}) - f(\mathbf{x}_0)|^\theta \leq L \|\mathbf{x}^*\| \quad \forall \mathbf{x} \in V, \mathbf{x}^* \in \partial^F f(\mathbf{x}),$$

where a convention $0^0 = 1$ is used.

The number θ is called a Łojasiewicz exponent of the function f at the critical point \mathbf{x}_0 . We now have all elements necessary to establish the convergence results of DCA with subanalytic data. The following theorem is about the convergence of the sequences $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$.

Theorem 34.5 *Let us consider DC problem (P_{dc}) with $\alpha \in \mathbb{R}$. Suppose that the sequences $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$ are defined by the DCA.*

- (1) Suppose that the DC function $f := g - h$ is subanalytic such that $\text{dom} f$ is closed; $f|_{\text{dom} f}$ is continuous; and around every critical point of (P_{dc}) , either g or h is differentiable with locally Lipschitz derivative. Assume that $\rho := \rho(g) + \rho(h) > 0$, where $\rho(g)$ and $\rho(h)$ are modulus of the strong convexity of g and h , respectively. If either the sequence $\{\mathbf{x}^k\}$ or $\{\mathbf{y}^k\}$ is bounded, then $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$ are convergent to critical points of (P_{dc}) and (D_{dc}) , respectively.
- (2) Similarly, the dual problem has the counterpart: suppose that $h^* - g^*$ is subanalytic such that $\text{dom}(h^* - g^*)$ is closed, $(h^* - g^*)|_{\text{dom}(h^* - g^*)}$ is continuous, and around critical point of (D_{dc}) , either g^* or h^* is differentiable with locally Lipschitz derivative. If $\rho(g^*) + \rho(h^*) > 0$ ($\rho(g^*)$ and $\rho(h^*)$ are modulus of the strong convexity of g^* and h^* , respectively) and either the sequence $\{\mathbf{x}^k\}$ or $\{\mathbf{y}^k\}$ is bounded, then $\{\mathbf{x}^k\}$ and $\{\mathbf{y}^k\}$ are convergent to critical points of (P_{dc}) and (D_{dc}) , respectively.

Corollary 34.4 Suppose that $g - h$ and $h^* - g^*$ are subanalytic functions with closed domain such that $(g - h)|_{\text{dom}(g-h)}$ and $(h^* - g^*)|_{\text{dom}(h^*-g^*)}$ are continuous. Assume that $\rho(g) + \rho(h) > 0$ as well as $\rho(g^*) + \rho(h^*) > 0$. If either the sequence $\{\mathbf{x}^k\}$ or $\{\mathbf{y}^k\}$ is bounded, then these sequences converge to critical points of (P_{dc}) and (D_{dc}) , respectively.

We next state a theorem about the convergence rate of $\{\mathbf{x}^k\}$ to its limit point \mathbf{x}^∞ .

Theorem 34.6 Suppose that the assumptions of Theorem 34.5(1) are satisfied. Let \mathbf{x}^∞ be the limit point of $\{\mathbf{x}^k\}$ at which the Łojasiewicz exponent $\theta \in [0, 1)$ of the function f is given. Then, there exist constants $\tau_1, \tau_2 > 0$ such that, for all $k \in \mathbb{N}$,

$$\|\mathbf{x}^k - \mathbf{x}^\infty\| \leq \sum_{j=k}^{\infty} \|\mathbf{x}^j - \mathbf{x}^{j+1}\| \leq \tau_1 \|\mathbf{x}^k - \mathbf{x}^{k-1}\| + \tau_2 \|\mathbf{x}^k - \mathbf{x}^{k-1}\|^{\frac{1-\theta}{\theta}}.$$

As a result, one has:

- If $\theta \in (1/2, 1)$, then $\|\mathbf{x}^k - \mathbf{x}^\infty\| \leq ck^{\frac{1-\theta}{1-2\theta}}$ for some $c > 0$.
- If $\theta \in (0, 1/2]$, then $\|\mathbf{x}^k - \mathbf{x}^\infty\| \leq cq^k$ for some $c > 0$ and $q \in (0, 1)$.
- If $\theta = 0$, then $\{\mathbf{x}^k\}$ is convergent in a finite number of steps.

Recent Developments and Open Problems in DC Programming and DCA

So far DCA seems to be the sole existing algorithm in nonsmooth nonconvex programming; the method is surely expected to continue to thrive. We summarize here some recent developments and open problems in DC programming and DCA:

1. Finding reformulations and DC decompositions well adapted to DC programs.
2. Exact penalty with/without error bounds in general DC programs, i.e., DC programs with DC constraints.
3. Using Nesterov smoothing techniques applied to generated convex subprograms for accelerating DCA, similar to the fast iterative shrinkage-thresholding algorithm.
4. Using proximal approaches techniques for both regularization and decomposition and improving DCA's convergence speed.
5. Using line search –standard DCA's step size being equal to 1– for the DC objective function in DCA.
6. Estimating DCA's convergence rate for special classes of DC programs.
7. DCA's global convergence for special DC programs, for example, DC programs with subanalytic data using the Łojasiewicz inequality for nonsmooth subanalytic functions.
8. Strategy for choosing initial points, taking into account specific structures of DC programs.
9. Composite functions techniques in DC programming and DCA.
10. Finding convex minorants of DC functions for computing lower bounds of optimal values.
11. Does DCA generate all existing algorithms for convex programming?

34.2.3 Links Between DCA and Standard Convex/Nonconvex Programming Approaches

This subsection discusses the links between DCA and majorization-minimization (MM), the successive convex approximation (SCA) approaches. Furthermore, some convex/nonconvex programming approaches are shown as special versions of DCA.

DCA Versus the MM and SCA Approaches

The general principle behind the MM was first introduced in the early work [36] by Ortega and Rheinboldt, followed by the work of de Leeuw [37]. Hunter and Lange named the method MM for the first time in [38]. The MM, like DCA, is a philosophy rather than a specific algorithm. Specifically, for solving the minimization problem

$$\min\{f(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \subset \mathbb{R}^n\},$$

the MM constructs a majorization function ϑ over $\mathcal{X} \times \mathcal{X}$ such that

$$f(\mathbf{x}) \leq \vartheta(\mathbf{x}, \mathbf{y}), \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$$

and

$$f(x) = \vartheta(x, x), \forall x \in \mathcal{X}.$$

At the iteration k , the MM updates

$$\mathbf{x}^{k+1} \in \arg \min \{ \vartheta(\mathbf{x}, \mathbf{x}^k) : \mathbf{x} \in \mathcal{X} \}.$$

In the literature, there are four typical approaches used to construct the surrogate: Jensen's inequality, convexity inequality, Cauchy-Schwartz inequality, and inequality of arithmetic and geometric means. When ϑ is convex, the MM, more recently, is also referred to as SCA. The idea of the MM is quite general, and it leaves us with a challenging task which is how to choose the appropriate surrogate. Working directly on the objective function f is also a source of difficulty of the MM. When it comes to DC programming context, DCA gives the simplest and the most closed convex surrogate function of f that benefits us in many aspects.

Regardless of its generality, most related works using the MM/SCA method in the literature can be recovered as versions of DCA based on latent DC structures of the problems. To demonstrate this statement, we consider here several ways to choose the convex surrogate in SCA.

Linear Upper Bounds of Concave Functions

When f is concave differentiable, the following affine function is often used as the surrogate of f :

$$\vartheta(\mathbf{x}, \mathbf{x}^k) = f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle,$$

and the SCA solve the following program to get \mathbf{x}^{k+1} :

$$\min \{ f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle : \mathbf{x} \in \mathcal{X} \}.$$

Obviously, this algorithm is DCA for the following DC program:

$$\min \{ g(\mathbf{x}) - h(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \},$$

where $g(\mathbf{x}) = 0$, $h(\mathbf{x}) = -f(\mathbf{x})$ that are both convex.

Quadratic Upper Bounds of Smooth Functions

When f is twice differentiable, the following upper bound is usually employed:

$$\vartheta(\mathbf{x}, \mathbf{x}^k) = f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{1}{2}(\mathbf{x} - \mathbf{x}^k)^T \mathbf{H}(\mathbf{x} - \mathbf{x}^k),$$

where \mathbf{H} is a positive semidefinite matrix such that $\mathbf{H} - \nabla^2 f(\mathbf{x})$ is also positive semidefinite.

This algorithm is a version of DCA applied to the following DC function:

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{H} \mathbf{x} - \left(\frac{1}{2} \mathbf{x}^T \mathbf{H} \mathbf{x} - f(\mathbf{x}) \right).$$

34.2.4 Proximal Upper Bounds of Convex Functions

The proximal upper bound of f ,

$$\vartheta(\mathbf{x}, \mathbf{x}^k) = f(\mathbf{x}) + \frac{\rho}{2} \|\mathbf{x} - \mathbf{x}^k\|^2,$$

is frequently used to gain the strong convexity of f . The corresponding SCA algorithm is known as the proximal point algorithm in convex programming. This algorithm, in the more specific setting, will be shown below as a version of DCA.

Proximal Point Algorithm for Convex Programming Is a DCA Version

Let $f \in \Gamma_0(X)$ and C be a nonempty closed convex set in X . Consider the following convex program:

$$\alpha = \inf \{ f(\mathbf{x}) : \mathbf{x} \in C \},$$

or equivalently,

$$\alpha = \inf \{ \chi_C(\mathbf{x}) + f(\mathbf{x}) : \mathbf{x} \in X \}.$$

This convex program can be considered as a DC program by the following DC decomposition:

$$\left(\chi_C(\mathbf{x}) + \frac{1}{2} \lambda \|\mathbf{x}\|^2 + f(\mathbf{x}) \right) - \frac{1}{2} \lambda \|\mathbf{x}\|^2,$$

where $\lambda > 0$.

The DCA iterates

$$\mathbf{x}^{k+1} \in \arg \min \left\{ f(\mathbf{x}) + \frac{1}{2} \lambda \|\mathbf{x}\|^2 - \langle \mathbf{x}, \lambda \mathbf{x}^k \rangle \right\},$$

and we recover the proximal point algorithm.

Goldstein-Levitin-Polyak Projection Algorithm Is a DCA Version

Let λ be a positive number such that the function $h(\mathbf{x}) = \frac{1}{2} \lambda \|\mathbf{x}\|^2 - f(\mathbf{x})$ is convex, and let $g(\mathbf{x}) = \chi_C(\mathbf{x}) + \frac{1}{2} \lambda \|\mathbf{x}\|^2$. Hence g, h are DC components of $\chi_C + f$ and the DCA takes steps

$$\mathbf{y}^k = \lambda \mathbf{x}^k - \boldsymbol{\eta}^k, \boldsymbol{\eta}^k \in \partial f(\mathbf{x}^k);$$

$$\mathbf{x}^{k+1} = P_C \left(\mathbf{x}^k - \frac{1}{\lambda} \boldsymbol{\eta}^k \right).$$

This is exactly the Goldstein-Levitin-Polyak projection algorithm [39].

Iterative Shrinkage-Thresholding Algorithm Is a DCA Version

Iterative shrinkage-thresholding algorithm was first introduced in [40] and later developed in [21, 41]. The most general case considered [21] is

$$\min_{\mathbf{x}} \{f(\mathbf{x}) = g_1(\mathbf{x}) + g_2(\mathbf{x})\},$$

where g_1, g_2 are convex and g_2 is differentiable with L -Lipschitz gradient. ISTA proposed in [21] iteratively computes \mathbf{x}^{k+1} by solving

$$\min_{\mathbf{x}} \left\{ g_1(\mathbf{x}) + \frac{L}{2} \|\mathbf{x} - \left(\mathbf{x}^k - \frac{1}{L} \nabla g_2(\mathbf{x}^k) \right)\|^2 \right\}.$$

It is observed that this algorithm is a version of DCA with the following DC decomposition:

$$f(\mathbf{x}) = \left(g_1(\mathbf{x}) + \frac{L}{2} \|\mathbf{x}\|^2 \right) - \left(\frac{L}{2} \|\mathbf{x}\|^2 - g_2(\mathbf{x}) \right).$$

When the Lipschitz constant L is unknown, the iterative shrinkage-thresholding algorithm with backtracking step size may be employed. Likewise, this algorithm is a version of DCA with successive DC decomposition.

The EM Algorithm for Exponential Families Is a Version of DCA

In statistics and data science, the EM (expectation maximization) algorithm is a popular tool for statistical estimation with incomplete data. The algorithm was named in 1977 in the work by Arthur Dempster et al. [42], where they pointed out that the method had already been proposed many times in special circumstances by earlier authors. In fact, EM is an MM-type algorithm where the surrogate is constructed by Jensen's inequality. When the likelihood function is an exponential family, the EM algorithm is a version of DCA. Specifically, we consider the following form of the likelihood:

$$\mathcal{P}(\mathbf{x}, \mathbf{v}, \mathbf{w}|\boldsymbol{\theta}) = \frac{1}{Z(\mathbf{x}|\boldsymbol{\theta})} \exp \left(\sum_{i=1}^D \boldsymbol{\theta}_i m_i(\mathbf{x}, \mathbf{v}, \mathbf{w}) \right),$$

where \mathbf{x} is a vector observation, \mathbf{v} is a visible state, \mathbf{w} is a hidden variable, $m_i(\mathbf{x}, \mathbf{v}, \mathbf{w})$ is some real or binary function, and

$$Z(\mathbf{x}|\boldsymbol{\theta}) = \sum_{\mathbf{v}, \mathbf{w}} \exp \left(\sum_{i=1}^D \boldsymbol{\theta}_i m_i(\mathbf{x}, \mathbf{v}, \mathbf{w}) \right)$$

is the normalization term.

Given the data set X , the parameter $\boldsymbol{\theta}$ is chosen to maximize the log-likelihood function:

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{\mathbf{x} \in X} \log \sum_{\mathbf{w}} \mathcal{P}(\mathbf{x}, \mathbf{v}, \mathbf{w}|\boldsymbol{\theta}).$$

The EM algorithm for maximizing $\mathcal{L}(\boldsymbol{\theta})$ consists of two steps described as follows:

- E-step: Determine the lower bound $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^k)$ of $\mathcal{L}(\boldsymbol{\theta})$ by

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^k) &= \sum_{\mathbf{x}} \sum_{\mathbf{w}} \mathcal{P}(\mathbf{w}|\mathbf{x}, \mathbf{v}; \boldsymbol{\theta}^k) \log \mathcal{P}(\mathbf{x}, \mathbf{v}, \mathbf{w}|\boldsymbol{\theta}) \\ &\quad - \sum_{\mathbf{x}} \sum_{\mathbf{w}} \mathcal{P}(\mathbf{w}|\mathbf{x}, \mathbf{v}; \boldsymbol{\theta}^k) \log \mathcal{P}(\mathbf{w}|\mathbf{x}, \mathbf{v}; \boldsymbol{\theta}^k), \end{aligned}$$

where

$$\mathcal{P}(\mathbf{w}|\mathbf{x}, \mathbf{v}; \boldsymbol{\theta}^k) = \frac{\exp \left(\sum_{i=1}^D \boldsymbol{\theta}_i^k m_i(\mathbf{x}, \mathbf{v}, \mathbf{w}) \right)}{\sum_{\mathbf{w}'} \exp \left(\sum_{i=1}^D \boldsymbol{\theta}_i^k m_i(\mathbf{x}, \mathbf{v}, \mathbf{w}') \right)}.$$

- M-step: Compute $\boldsymbol{\theta}^{k+1}$ by maximizing $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^k)$.

It can be shown that this EM algorithm is DCA for minimizing $-\mathcal{L}(\boldsymbol{\theta})$ with the DC decomposition $-\mathcal{L}(\boldsymbol{\theta}) = g - h$, where

$$\begin{aligned} g(\boldsymbol{\theta}) &= \sum_{\mathbf{x} \in X} \log \sum_{\mathbf{v}, \mathbf{w}} \exp \left(\sum_{i=1}^D \boldsymbol{\theta}_i m_i(\mathbf{x}, \mathbf{v}, \mathbf{w}) \right), \\ h(\boldsymbol{\theta}) &= \sum_{\mathbf{x} \in X} \log \sum_{\mathbf{w}} \exp \left(\sum_{i=1}^D \boldsymbol{\theta}_i m_i(\mathbf{x}, \mathbf{v}, \mathbf{w}) \right). \end{aligned}$$

The Concave-Convex Procedure Is an Instance of DCA in Smooth Optimization

The concave-convex procedure was first proposed in 2003 [43], where the authors assumed that the objective function f is twice differentiable with the DC decomposition

$$f(\mathbf{x}) = f_{\text{vex}}(\mathbf{x}) + f_{\text{cav}}(\mathbf{x}),$$

and the feasible set C is defined by linear constraints. Under these assumptions, the concave-convex procedure takes steps

$$\mathbf{x}^{k+1} \in \arg \min \{f_{\text{vex}}(\mathbf{x}) + \langle \mathbf{x}, \nabla f_{\text{cav}}(\mathbf{x}^k) \rangle : \mathbf{x} \in C\}.$$

Different from other mentioned algorithms where the DC structure of the problems is somehow hidden, the concave-convex procedure is easily recognized as nothing but the DCA.

34.3 Stochastic Optimization in the Literature

Although deterministic programming is able to cover a large class of optimization problems in our real world, such a modeling framework would find it difficult to cope with the presence of randomness. The randomness appears unavoidably since many real-world systems are very complicated, for instance, one input can lead to many possible outcomes, relations between parameters are unclear, or data is uncertain. In these cases, stochastic programming would be a better choice to fit the system. Sometimes, one can nevertheless simplify the problems to make them deterministic in order to use powerful tools of deterministic optimization. However, stochastic programming is more suitable for representing the true nature of the problems and preserving as much as possible the information we are given. In the context of stochastic programming, objective and constraints can be uncertain in various forms such as expectation, conditional expectation, probability, etc. Therefore, we are quite flexible to manipulate the models to describe reasonably our real-life problems. We refer the reader to [44] for some important problems such as the new vendor problem, portfolio selection problem, supply chain network design problem, multiproduct assembly problem, etc. These problems could be one-stage, two-stage, or multistage, with or without chance constraints.

In addition, when solving deterministic programming problems by using multistage algorithms, there is an error ϵ_k at each step. This error is unknown in general, so it can be viewed as a random quantity. Hence, the randomness again appears even in the deterministic programming context [45–51].

It is noteworthy that, among the general class of stochastic optimization algorithms, there is a special class of algorithms whose stochastic approximations are constructed by new fresh samples from the interested distribution. That is to say, the update rules of these stochastic algorithms are performed by using streaming data coming from a certain (possibly unknown) distribution. In this chapter, we refer to these kinds of stochastic algorithms as *online stochastic*.

In the literature of stochastic optimization, it can be traced back to the pioneering work by Robbins and Monro [52], where the authors designed a (nonstationary) Markov chain for solving stochastically the equation

$$M(x) = a,$$

where M is unknown by nature. The Markov chain is constructed in a way that, under some assumptions, it converges in probability to a unique solution θ of the above equation.

The idea of this chapter has inspired many researchers to adopt this method for various challenging problems, especially for those that arise in optimization. It has been followed

by many stochastic methods for solving deterministic or stochastic programs.

Stochastic optimization, in a same way as deterministic optimization, has been developed highly in convex or strongly convex programming. Here we present some of these algorithms.

The stochastic subgradient method for minimizing a function F ,

$$\mathbf{w}^{k+1} \leftarrow \mathbf{w}^k - \alpha_k (\mathbf{y}^k + \boldsymbol{\xi}_k),$$

where $\mathbf{y}^k \in \partial F(\mathbf{w}^k)$ and $\boldsymbol{\xi}_k$ is some random noise satisfying certain conditions. This method generalizes stochastic gradient descent. The explicit form and computational advantages of these two algorithms make them the objects of many works and applications [53–58]. Besides, some variants of these methods which pay attention to second-order information but still remain robust for large-scale setting are studied [59–67]. These works mainly focus on smooth programming since the methods require second-order information. Furthermore, the problems can be convex or nonconvex, but convexity is more favorable. These algorithms are based on the classical quasi-Newton method combining with stochastic approximation idea. Broadly, these algorithms take steps

$$\mathbf{w}^{k+1} \leftarrow \mathbf{w}^k - \alpha_k \mathbf{H}_k \times (\mathbf{y}^k + \boldsymbol{\xi}_k),$$

where $\mathbf{y}^k + \boldsymbol{\xi}_k$ is some stochastic quantity approximating to $\nabla F(\mathbf{w}^k)$ and \mathbf{H}_k is an appropriate deterministic or stochastic matrix approximating to the inverse of the Hessian matrix $\nabla^2 F(\mathbf{w}^k)$.

When it comes to large-sum problem, it is worth noting that the sum of deterministic functions can be rewritten in the expectation form as follows:

$$F(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n f_i(\mathbf{w}) = E(f_I(\mathbf{w})),$$

where I is a uniform random variable over the set $\{1, 2, \dots, n\}$.

This kind of function has been of great concern. It appears frequently in various contexts of applied mathematics. Regarding its special structure, many stochastic methods have been developed to handle this class of problems. We name some well-known algorithms in this context. Roux et al. [68] proposed an algorithm called stochastic average gradient that can be viewed as a stochastic version of incremental average gradient [69]. Stochastic average gradient needs to store n gradients of the previous iteration. After drawing a random index I_k from $\{1, 2, \dots, n\}$, the algorithm updates

$$\mathbf{w}^k \leftarrow \mathbf{w}^{k-1} - \frac{\alpha_k}{n} \sum_{i=1}^n \mathbf{y}_i^k,$$

with

$$y_i^k = \begin{cases} f_i'(\mathbf{w}^{k-1}) & \text{if } i = I_k \\ y_i^{k-1} & \text{otherwise.} \end{cases}$$

Ghadimi and Lan [70] proposed an algorithm named the accelerated stochastic approximation. The algorithm is inspired by accelerated algorithm of Nesterov [71]. The idea is replacing the gradients in Nesterov's algorithm by stochastic gradients or subgradients.

To name more works in the similar stochastic approximation manner, we refer to stochastic dual coordinate ascent and its proximal versions [72–74], stochastic variance reduced gradient and its proximal variants [75–77], stochastic mirror descent [78–80], etc.

In nonconvex programming, Mairal [81] designed a stochastic version, called MISO, of the basic majorization-minimization scheme by exploiting the structure of the objective. At the iteration k , instead of choosing a surrogate g_k for F near \mathbf{w}^{k-1} , the algorithm constructs a surrogate g_k^l for one random element f_l . The surrogate g_k^l is then combined with information of the previous iteration to construct a stochastic surrogate for F near \mathbf{w}^{k-1} . The minimization process is then applied to this stochastic surrogate to get \mathbf{w}^k .

Recently, a new class of stochastic model-based algorithms has been developed for stochastic composite function

$$F(\mathbf{w}) = E(h(c(\mathbf{w}, \mathbf{Z}), \mathbf{Z})) = \int_{\mathcal{S}} h(c(\mathbf{w}, s), s) dP(s),$$

where the function $\mathbf{w} \mapsto h(\mathbf{w}, s)$ is closed convex and the function $\mathbf{w} \mapsto c(\mathbf{w}, s)$ is continuous for each $s \in \mathcal{S}$.

The minimization problem corresponding to this objective is a special case of stochastic weakly convex optimization [82, 83]:

$$\underset{\mathbf{w} \in X}{\text{minimize}} F(\mathbf{w}) + \varphi(\mathbf{w}) = E(f(\mathbf{w}, \mathbf{Z})) + \varphi(\mathbf{w}),$$

where $X \subset \mathbb{R}^d$ is a closed convex set, φ is a closed convex function, and $f(\cdot, s)$ is weakly convex for each s in \mathcal{S} .

The model-based strategy computes \mathbf{w}^{k+1} by solving the following problem:

$$\underset{\mathbf{y} \in X}{\text{minimize}} \left\{ f_{\mathbf{w}^k}(\mathbf{y}, \mathbf{Z}_k) + \varphi(\mathbf{y}) + \frac{1}{2\alpha_k} \|\mathbf{y} - \mathbf{w}^k\|^2 \right\},$$

where $\mathbf{Z}_k \sim \mathbf{Z}$, $\alpha_k > 0$ is a step size and $f_{\mathbf{w}^k}(\cdot, s)$ is an approximation, or a model, of $f(\cdot, s)$ at \mathbf{w}^k .

With different choices of the model $f_{\mathbf{w}^k}(\cdot, s)$, the mentioned stochastic model-based algorithm can recover many classical stochastic algorithms including stochastic subgradient method, stochastic prox-linear method, stochastic proximal point method, and guarded stochastic proximal point method [82].

Regarding the outperformance of DCA in deterministic nonconvex programming, it is natural to develop stochastic versions for DCA. This idea is the subject of some recent works. However, such kinds of works remain to be rare. Le Thi et al. [84] proposed a stochastic DCA scheme for the large sum of nonconvex functions problem:

$$F(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n f_i(\mathbf{w}) + \lambda p(\mathbf{w}),$$

where p is DC, $p = \tilde{g} - \tilde{h}$, and f_i is differentiable with L -Lipschitz gradient. The problem can be reformulated as the following stochastic programming problem:

$$\underset{\mathbf{w}}{\text{minimize}} \{F(\mathbf{w}) = G(\mathbf{w}) - H(\mathbf{w})\}, \quad (34.3)$$

where $H(\mathbf{w}) = E(h_I(\mathbf{w}))$ and I is a uniform random variable over $\{1, 2, \dots, n\}$. The proposed algorithm modifies the traditional DCA by approximating stochastically the subgradient $\partial H(\mathbf{w})$.

Le Thi et al. [85] developed the idea of [84] to encounter directly the general nonconvex large-sum problem

$$F(\mathbf{w}) = G(\mathbf{w}) - \frac{1}{n} \sum_{i=1}^n h_i(\mathbf{w}).$$

It is observed that this problem can be written in the form of (34.3).

The authors introduced two stochastic DCA schemes for solving this problem. The first scheme replaces deterministic subgradient $\partial H(\mathbf{w})$ by its stochastic estimator and proceeds the same as traditional DCA, while the second scheme only requires to compute inexactly stochastic estimator of $\partial H(\mathbf{w})$ and solve inexactly subproblems.

Le Thi et al. [86] introduced seven stochastic DCA schemes for minimizing directly the expected loss with the general underlying distribution \mathbf{Z} :

$$F(\mathbf{w}) = E(g(\mathbf{w}, \mathbf{Z})) - E(h(\mathbf{w}, \mathbf{Z})) + r(\mathbf{w}).$$

Roughly speaking, g, h, r are satisfied some assumptions and F is DC. The mentioned seven algorithms consist of many forms such as approximating both ∂H and G or approximating only ∂H , the ways of updating stochastic subgradients, adding proximal terms to subproblems or not, etc.

Nitanda and Suzuki [87] gave a proximal stochastic DCA scheme for DC program $g(\mathbf{w}) - h(\mathbf{w})$ by using an unbiased estimator $v_h(\mathbf{w})$ for $\nabla h(\mathbf{w})$, where h is assumed to be differentiable. After that, the algorithm solves inexactly the following proximal convex program to get \mathbf{w}^{k+1} :

$$\begin{aligned} & \underset{\mathbf{w} \in \mathbb{R}^d}{\text{minimize}} \left\{ \phi_k(\mathbf{w}) = g(\mathbf{w}) + \frac{1}{2} \|\mathbf{w} - \mathbf{w}^k\|_{\mathbf{H}_k}^2 \right. \\ & \left. - (h(\mathbf{w}^k) + \langle v_h(\mathbf{w}^k), \mathbf{w} - \mathbf{w}^k \rangle) \right\}, \end{aligned}$$

where \mathbf{H}_k is a positive definite matrix and the norm $\|\cdot\|_{\mathbf{H}_k}$ is Mahalanobis norm with respect to \mathbf{H}_k .

Similar ideas were adopted by Xu et al. [88].

Liu et al. [89] proposed an online stochastic DCA scheme called RCS algorithm for solving a special class of two-stage stochastic programs with a linearly bi-parameterized recourse function defined by a convex quadratic program. Broadly, the authors approximated both components G and ∇H by new samples and added the proximal term $\frac{1}{2\gamma} \|\mathbf{w} - \mathbf{w}^k\|^2$ to subproblems. The RCS algorithm was then analyzed in the context of the mentioned two-stage stochastic programs. Under some assumptions, every limit point of the sequence generated by RCS algorithm is a generalized critical point, or a d-stationary point of the problem.

It is worth noting that, in the context of nonconvex optimization, some stochastic algorithms are based on DCA, but the authors are not aware of this fact. For example, Allen-Zhu et al. [90] proposed a stochastic variance reduction algorithm for a sum of nonconvex functions with L -Lipschitz continuous gradient (L -smooth). As a variance reduction method, the proposed algorithm consists of two loops. The inner loop takes steps

$$\mathbf{w}_s^{k+1} \leftarrow \mathbf{w}_s^k - \eta (\nabla f_i(\mathbf{w}_s^k) - \nabla f_i(\mathbf{w}_s^0) + \nabla f(\mathbf{w}_s^0)),$$

where s is the index of the outer loop, i is a uniformly random choice in $\{1, 2, \dots, n\}$, and $\eta \leq 1/L$ is a learning rate.

We can see that, with the L -smooth property, f becomes DC with the following decomposition:

$$f(\mathbf{w}) = \frac{1}{2\eta} \|\mathbf{w}\|^2 - \left(\frac{1}{2\eta} \|\mathbf{w}\|^2 - f(\mathbf{w}) \right),$$

where $\eta \leq 1/L$.

The DCA iterates

$$\mathbf{w}^{k+1} \leftarrow \mathbf{w}^k - \eta \nabla f(\mathbf{w}^k).$$

It is observed that the update term in the Allen-Zhu's algorithm can be viewed as a stochastic approximation of $\nabla f(\mathbf{w}_s^k)$, so this algorithm can be referred to as a stochastic version of DCA.

In this chapter, we design two stochastic versions of DCA in an online manner. Our setting is quite general in terms of the distribution of \mathbf{Z} which does not need to be discrete. The concave part of our DC decomposition does not need to be differentiable. The first scheme stochastically approximates a subgradient of H across iterations. Furthermore, our second

online stochastic algorithm approximates stochastically not only the subgradient ∂H but also the convex part G , which let us be able to handle situations where the distribution of \mathbf{Z} is unknown. However, unlike algorithms mentioned in [87, 88], our algorithms require to solve exactly subproblems for guaranteeing the almost sure convergence in our analysis. Our algorithms are distinct from [86] by the ways of approximating subgradients.

34.4 Stochastic DC Programming and DCA

This section introduces a class of stochastic DC programs and proposes two stochastic DCA schemes for solving this class. Our stochastic DCA schemes are designed in an online manner, so we name them *online stochastic DCA*.

From now on, the reader may be aware that most of the statements are understood as almost sure (a.s), i.e., they are true with probability 1.

34.4.1 A Class of Stochastic DC Programs

This section introduces a class of stochastic DC programs that we study in this chapter.

Let (Ω, \mathcal{M}, P) be a complete probability space and \mathbf{Z} be a random vector, $\mathbf{Z} : \Omega \rightarrow \mathbb{R}^n$.

Let $\Xi = \text{supp}(P_{\mathbf{Z}})$ be a support of the probability measure $P_{\mathbf{Z}}$, where $P_{\mathbf{Z}}$, the probability measure on \mathbb{R}^n , is the distribution of \mathbf{Z} .

By definition, a point $\mathbf{x} \in \mathbb{R}^n$ is in $\text{supp}(P_{\mathbf{Z}})$ if $P_{\mathbf{Z}}(N_{\mathbf{x}}) > 0$ for every neighborhood $N_{\mathbf{x}}$ of \mathbf{x} , where the topology used to define neighborhood is the Euclidean topology of \mathbb{R}^n .

A basic property of the support Ξ is that it is closed in \mathbb{R}^n . Moreover, $P_{\mathbf{Z}}(\Xi^c) = 0$ since \mathbb{R}^n is the topological Hausdorff space and $P_{\mathbf{Z}}$ is a Radon measure in \mathbb{R}^n .

Consider a function $f : \mathbb{R}^m \times \Xi \rightarrow \mathbb{R} \cup \{+\infty\}$. Since $\mathbb{R} \cup \{+\infty\}$ is a totally ordered set, there is a corresponding order topology τ_{∞} . Let $\mathcal{B}_{\infty} = \mathcal{B}(\tau_{\infty})$ be a Borel sigma-algebra generated by τ_{∞} . The function f is assumed to be measurable with respect to $(\mathcal{B}(\mathbb{R}^m) \otimes \mathcal{A}, \mathcal{B}_{\infty})$, where \mathcal{A} is a Borel sigma-algebra relative to Ξ .

The effective domain of f is assumed to be

$$\text{dom} f := \{(\mathbf{x}, \xi) \in \mathbb{R}^m \times \Xi \mid f(\mathbf{x}, \xi) < +\infty\} = S \times \Xi,$$

where $S \subset \mathbb{R}^m$ is a nonempty, compact, and convex set.

Suppose that for all $\mathbf{x} \in S$, $\int_{\Xi} |f(\mathbf{x}, \xi)| dP_{\mathbf{Z}} < +\infty$.

Let $\varphi(\mathbf{x}) = \int_{\Xi} f(\mathbf{x}, \xi) dP_{\mathbf{Z}}$, $\forall \mathbf{x} \in S$. The optimization problem considered is

$$\text{minimize } \varphi(\mathbf{x}), \mathbf{x} \in S.$$

The function f is extended to the entire space as $\bar{f} : \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$,

$$\bar{f}(\mathbf{x}, \boldsymbol{\xi}) = \begin{cases} f(\mathbf{x}, \boldsymbol{\xi}) & \text{if } (\mathbf{x}, \boldsymbol{\xi}) \in S \times \Xi, \\ +\infty & \text{otherwise.} \end{cases}$$

It is verified that this extension reserves measurability, i.e., \bar{f} is $(\mathcal{B}(\mathbb{R}^m) \otimes \mathcal{B}(\mathbb{R}^n), \mathcal{B}_\infty)$ measurable.

Suppose that \bar{f} can be written as $\bar{f} = g - h$, where $g, h : \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ are Borel measurable functions with $\text{dom } g = \text{dom } h = S \times \Xi$.

The first assumption of g and f is formally stated as the following:

Assumption 1

- (1) For all $\mathbf{z} \in \Xi$, $g(\cdot, \mathbf{z})$ and $h(\cdot, \mathbf{z})$ are convex, lower semicontinuous.
- (2) For all $\mathbf{w} \in S$, $g(\mathbf{w}, \mathbf{Z}), h(\mathbf{w}, \mathbf{Z}) \in L^1(\Omega)$.
- (3) For all $\mathbf{z} \in \Xi$, $\text{dom } \partial h(\cdot, \mathbf{z}) = S$.
- (4) There exists a Borel measurable selector

$$\tau : \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}^m$$

such that

$$\forall \mathbf{w} \in S, \mathbf{z} \in \Xi, \tau(\mathbf{w}, \mathbf{z}) \in \partial_{\mathbf{w}} h(\mathbf{w}, \mathbf{z}).$$

- (5) There exists a Borel measurable function

$$\tilde{\tau} : \mathbb{R}^n \rightarrow \mathbb{R}$$

such that $\int_{\Xi} |\tilde{\tau}(\mathbf{z})| \, dP_{\mathbf{Z}} < \infty$ and

$$\text{for all } \mathbf{w} \in S, \mathbf{z} \in \Xi, \|\tau(\mathbf{w}, \mathbf{z})\| \leq \tilde{\tau}(\mathbf{z}).$$

- (6) For all $\mathbf{w} \in S$, $E(\|\tau(\mathbf{w}, \mathbf{Z})\|^2) < +\infty$. □

Assumptions 1(1) and 1(2) are merely regular assumptions, and Assumption 1(3) means that subgradient vectors of $h(\cdot, \mathbf{z})$ do not explode at the boundary of the domain. Assumption 1(4) is that we can choose a measurable selector from the beginning, and then an algorithm will choose subgradients following this selector. Assumptions 1(5) and 1(6) are for the boundedness of the selector.

Under Assumption 1, let $G(\mathbf{w}) = E(g(\mathbf{w}, \mathbf{Z}))$ and $H(\mathbf{w}) = E(h(\mathbf{w}, \mathbf{Z}))$; then $\text{dom } G = \text{dom } H = S$ and G, H are convex. Let $C = G - H$; then $\text{dom } C = S$ and the optimization problem becomes

$$\text{minimize } C(\mathbf{w}), \mathbf{w} \in \mathbb{R}^m.$$

We now add some assumptions for G and H .

Assumption 2

- (1) G, H are lower semicontinuous.
- (2) $\text{dom } \partial H = S$ and $\rho(H) > 0$.
- (3) $\sup_{\mathbf{w} \in S} |C(\mathbf{w})| < +\infty$. □

Assumption 2(1) is for the DC form of the objective function C . In fact, Assumption 2(2) can be induced by Assumption 1(1) with some additional conditions. For example, if $g(\mathbf{w}, \mathbf{Z})$ are uniform bounded below by a L^1 random variable, then Assumption 2(1) is verified by Fatou's lemma. Assumption 2(2) can be interpreted that subgradient vectors of H do not explode at the boundary of the domain. Finally, Assumption 2(3) states that C is nowhere going to infinity.

34.4.2 Online Stochastic DCA1

This subsection proposes the first online stochastic DCA scheme for solving the stochastic DC problem introduced in Sect. 34.4.1.

Firstly, let us define a function denoted by ϑ as the following:

$$\begin{aligned} \vartheta : \mathbb{R}^m \times \mathbb{R}^m &\rightarrow \mathbb{R} \cup \{+\infty\} \\ (\mathbf{w}, \mathbf{t}) &\mapsto G(\mathbf{w}) - \langle \mathbf{w}, \mathbf{t} \rangle. \end{aligned}$$

We can see that ϑ is Borel measurable and lower semicontinuous.

Since $G - \langle \cdot, \mathbf{t} \rangle$ is lower semicontinuous and S is compact, it follows that $\arg \min_{\mathbf{w} \in \mathbb{R}^m} \{G(\mathbf{w}) - \langle \mathbf{w}, \mathbf{t} \rangle\}$ is nonempty, for all $\mathbf{t} \in \mathbb{R}^m$.

By applying selection theorem [91], there exists a Borel measurable function $\kappa : \mathbb{R}^m \rightarrow \mathbb{R}^m$ such that $\kappa(\mathbf{t}) \in \arg \min_{\mathbf{w} \in \mathbb{R}^m} \{G(\mathbf{w}) - \langle \mathbf{w}, \mathbf{t} \rangle\}$, for all $\mathbf{t} \in \mathbb{R}^m$.

We now give the online stochastic DCA scheme called online stochastic DCA1 as follows.

Algorithm 1 Online stochastic DCA1

Step 1. Choose $\mathbf{w}^0 \in S$.

Step 2. For $k = 0, 1, 2, \dots$ do

1. Draw independently $\mathbf{Z}_{k,1}, \mathbf{Z}_{k,2}, \dots, \mathbf{Z}_{k,n_k}$ from \mathbf{Z} in such a way that they are also independent to the past.
2. Compute $\mathbf{t}^k = \frac{1}{n_k} \sum_{i=1}^{n_k} \tau(\mathbf{w}^k, \mathbf{Z}_{k,i})$.
3. Solve the convex program to get \mathbf{w}^{k+1} :

$$\mathbf{w}^{k+1} = \kappa(\mathbf{t}^k) \in \arg \min_{\mathbf{w} \in \mathbb{R}^m} \{G(\mathbf{w}) - \langle \mathbf{t}^k, \mathbf{w} \rangle\}.$$

4. If stopping criteria are satisfied, break. □

It is clear that online stochastic DCA1 is well-defined. Moreover, the sequence created by the algorithm is contained in S .

For short, we denote $\mathbf{Z}_k = \mathbf{Z}_{k,1:n_k}$ for every $k \in \mathbb{N}$ and let $\mathcal{P}_k = \sigma(\mathbf{Z}_0, \mathbf{Z}_1, \dots, \mathbf{Z}_{k-1})$. By the Borel measurability of τ and κ , it follows by induction that \mathbf{w}^k is \mathcal{P}_k measurable, for all $k \geq 1$, and \mathbf{t}^k is \mathcal{P}_{k+1} measurable, for all $k \geq 0$.

Lemma 34.1 *Let $0 < \bar{\rho} < \rho(H)$. Under assumptions 1,2, the following inequality holds for all $k \in \mathbb{N}$:*

$$E(C(\mathbf{w}^{k+1})|\mathcal{P}_k) \leq -\frac{\bar{\rho}}{2}E(\|\mathbf{w}^k - \mathbf{w}^{k+1}\|^2|\mathcal{P}_k) + C(\mathbf{w}^k) + \frac{V_Z(\tau(\mathbf{w}^k, \mathbf{Z}))^{\frac{1}{2}}}{\sqrt{n_k}}E(\|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2|\mathcal{P}_k)^{\frac{1}{2}},$$

where

$$V_Z(\tau(\mathbf{w}^k, \mathbf{Z})) = E_Z(\|\tau(\mathbf{w}^k, \mathbf{Z})\|^2) - \|E_Z(\tau(\mathbf{w}^k, \mathbf{Z}))\|^2.$$

Proof. Since $v(\mathbf{w}^k) \in \partial_w H(\mathbf{w}^k)$, it follows that

$$H(\mathbf{w}^{k+1}) \geq H(\mathbf{w}^k) + \langle v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle + \frac{\bar{\rho}}{2}\|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2. \quad (34.4)$$

By the definition of \mathbf{w}^{k+1} , it follows that

$$G(\mathbf{w}^k) \geq G(\mathbf{w}^{k+1}) + \langle \mathbf{t}^k, \mathbf{w}^k - \mathbf{w}^{k+1} \rangle. \quad (34.5)$$

From (34.4) and (34.5),

$$\begin{aligned} G(\mathbf{w}^{k+1}) - H(\mathbf{w}^{k+1}) &\leq G(\mathbf{w}^k) - H(\mathbf{w}^k) \\ &\quad + \langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle \\ &\quad - \frac{\bar{\rho}}{2}\|\mathbf{w}^k - \mathbf{w}^{k+1}\|^2, \end{aligned}$$

or equivalently

$$C(\mathbf{w}^{k+1}) \leq C(\mathbf{w}^k) - \frac{\bar{\rho}}{2}\|\mathbf{w}^k - \mathbf{w}^{k+1}\|^2 + \langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle. \quad (34.6)$$

By Assumptions 1(5) and 2(3), $C(\mathbf{w}^k)$, $\langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle$ belong to $L^1(\Omega)$. By taking conditional expectation of (34.6) with respect to \mathcal{P}_k , we get

$$\begin{aligned} E(C(\mathbf{w}^{k+1})|\mathcal{P}_k) &\leq E(\langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle|\mathcal{P}_k) + C(\mathbf{w}^k) \\ &\quad - \frac{\bar{\rho}}{2}E(\|\mathbf{w}^k - \mathbf{w}^{k+1}\|^2|\mathcal{P}_k). \end{aligned}$$

By applying Schwartz inequality and Holder inequality, we get

$$\begin{aligned} E(\langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle|\mathcal{P}_k) &\leq E(\|\mathbf{t}^k - v(\mathbf{w}^k)\| \\ &\quad \times \|\mathbf{w}^{k+1} - \mathbf{w}^k\||\mathcal{P}_k) \\ &\leq E(\|\mathbf{t}^k - v(\mathbf{w}^k)\|^2|\mathcal{P}_k)^{\frac{1}{2}} \\ &\quad E(\|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2|\mathcal{P}_k)^{\frac{1}{2}}. \end{aligned}$$

Therefore,

$$\begin{aligned} E(C(\mathbf{w}^{k+1})|\mathcal{P}_k) &\leq C(\mathbf{w}^k) - \frac{\bar{\rho}}{2}E(\|\mathbf{w}^k - \mathbf{w}^{k+1}\|^2|\mathcal{P}_k) \\ &\quad + E(\|\mathbf{t}^k - v(\mathbf{w}^k)\|^2|\mathcal{P}_k)^{\frac{1}{2}} \\ &\quad E(\|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2|\mathcal{P}_k)^{\frac{1}{2}}. \end{aligned}$$

We now calculate $E(\|\mathbf{t}^k - v(\mathbf{w}^k)\|^2|\mathcal{P}_k)$. By the independence of $\mathbf{Z}_{k,i}$, $\mathbf{Z}_{k,j}$ when $i \neq j$, it follows that

$$\begin{aligned} E(\|\mathbf{t}^k - v(\mathbf{w}^k)\|^2|\mathcal{P}_k) &= E\left(\left\|\frac{1}{n_k}\sum_{i=1}^{n_k}(\tau(\mathbf{w}^k, \mathbf{Z}_{k,i}) - v(\mathbf{w}^k))\right\|^2|\mathcal{P}_k\right) \\ &= \frac{1}{n_k^2}\sum_{i=1}^{n_k}E(\|\tau(\mathbf{w}^k, \mathbf{Z}_{k,i}) - v(\mathbf{w}^k)\|^2|\mathcal{P}_k). \end{aligned}$$

We have an observation that

$$\begin{aligned} E(\|\tau(\mathbf{w}^k, \mathbf{Z}_{k,i}) - v(\mathbf{w}^k)\|^2|\mathcal{P}_k) &= E(\|\tau(\mathbf{w}^k, \mathbf{Z}_{k,i})\|^2|\mathcal{P}_k) + E(\|v(\mathbf{w}^k)\|^2|\mathcal{P}_k) \\ &\quad - 2E(\langle \tau(\mathbf{w}^k, \mathbf{Z}_{k,i}), v(\mathbf{w}^k) \rangle|\mathcal{P}_k) \\ &= E_Z(\|\tau(\mathbf{w}^k, \mathbf{Z})\|^2) + \|v(\mathbf{w}^k)\|^2 \\ &\quad - 2\langle E_Z(\tau(\mathbf{w}^k, \mathbf{Z}_{k,i})|\mathcal{P}_k), v(\mathbf{w}^k) \rangle \\ &= E_Z(\|\tau(\mathbf{w}^k, \mathbf{Z})\|^2) + \|v(\mathbf{w}^k)\|^2 \\ &\quad - 2\langle E_Z(\tau(\mathbf{w}^k, \mathbf{Z})), v(\mathbf{w}^k) \rangle \\ &= E_Z(\|\tau(\mathbf{w}^k, \mathbf{Z})\|^2) - \|v(\mathbf{w}^k)\|^2. \end{aligned} \quad (34.7)$$

In the computation (34.7), the theorem in appendix A is applied. This result is classic and important in martingale theory, for example, we refer to [92] for its special case. We dedicate appendix A to its detail proof.

We get the result

$$\begin{aligned} \frac{1}{n_k^2}\sum_{i=1}^{n_k}E(\|\tau(\mathbf{w}^k, \mathbf{Z}_{k,i}) - v(\mathbf{w}^k)\|^2|\mathcal{P}_k) &= \frac{1}{n_k}(E_Z(\|\tau(\mathbf{w}^k, \mathbf{Z})\|^2) - \|v(\mathbf{w}^k)\|^2). \end{aligned}$$

Therefore, we achieve

$$\begin{aligned}
E(\|\mathbf{t}^k - \nu(\mathbf{w}^k)\|^2 | \mathcal{P}_k) &= \frac{1}{n_k} (E_Z(\|\tau(\mathbf{w}^k, \mathbf{Z})\|^2) - \|\nu(\mathbf{w}^k)\|^2) \\
&\leq \frac{V_Z(\tau(\mathbf{w}^k, \mathbf{Z}))}{2\bar{\rho} \cdot n_k}. \\
&= \frac{1}{n_k} (E_Z(\|\tau(\mathbf{w}^k, \mathbf{Z})\|^2) \\
&\quad - \|E_Z(\tau(\mathbf{w}^k, \mathbf{Z}))\|^2).
\end{aligned}$$

The variance is denoted as

$$V_Z(\tau(\mathbf{w}^k, \mathbf{Z})) := E_Z(\|\tau(\mathbf{w}^k, \mathbf{Z})\|^2) - \|E_Z(\tau(\mathbf{w}^k, \mathbf{Z}))\|^2,$$

and we get an inequality

$$\begin{aligned}
E(C(\mathbf{w}^{k+1}) | \mathcal{P}_k) &\leq -\frac{\bar{\rho}}{2} E(\|\mathbf{w}^k - \mathbf{w}^{k+1}\|^2 | \mathcal{P}_k) + C(\mathbf{w}^k) \\
&\quad + \frac{V_Z(\tau(\mathbf{w}^k, \mathbf{Z}))^{\frac{1}{2}}}{\sqrt{n_k}} E(\|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2 | \mathcal{P}_k)^{\frac{1}{2}}.
\end{aligned} \tag{34.8}$$

Assumption 3 (Variance is Bounded in Expectation)

There exists $M > 0$ such that

$$\forall k \in \mathbb{N}, E(V_Z(\tau(\mathbf{w}^k, \mathbf{Z}))) \leq M.$$

Theorem 34.7 Under the Assumptions 1, 2, 3, if the sequence of sample size $\{n_k\}$ satisfies

$$\sum_{k=1}^{\infty} \frac{1}{n_k} < \infty,$$

the iterates of online stochastic DCA1 satisfy:

(1) There exists $C^\infty \in L^1(\Omega)$ such that

$$C(\mathbf{w}^k) \rightarrow C^\infty \text{ a.s.}$$

(2) $\|\mathbf{w}^{k+1} - \mathbf{w}^k\| \rightarrow 0$ a.s.

(3) There exists a measurable set $\mathcal{L} \subset \Omega$ with $P(\mathcal{L}) = 1$ such that for each $\omega \in \mathcal{L}$, the sequence $\{\mathbf{w}^k(\omega)\}$ has a subsequence converging to a point $\mathbf{w}^*(\omega)$ which is a critical point of $C = G - H$.

Proof. (1) From (34.8), we apply AM-GM inequality to get

$$\begin{aligned}
E(C(\mathbf{w}^{k+1}) - C(\mathbf{w}^k) | \mathcal{P}_k) &\leq \frac{V_Z(\tau(\mathbf{w}^k, \mathbf{Z}))^{\frac{1}{2}}}{\sqrt{n_k}} E(\|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2 | \mathcal{P}_k)^{\frac{1}{2}} \\
&\quad - \frac{\bar{\rho}}{2} E(\|\mathbf{w}^k - \mathbf{w}^{k+1}\|^2 | \mathcal{P}_k)
\end{aligned}$$

By the Assumption 2(3), there exists R such that $C(\mathbf{w}) \geq R, \forall \mathbf{w} \in S$.

Let $D(\mathbf{w}) = C(\mathbf{w}) - R \geq 0$; we get

$$E(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k) | \mathcal{P}_k) \leq \frac{V_Z(\tau(\mathbf{w}^k, \mathbf{Z}))}{2\bar{\rho} \cdot n_k}.$$

Let $F_k = [E(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k) | \mathcal{P}_k) > 0]$,

$$\begin{aligned}
&\sum_{k=1}^{\infty} E(1_{F_k}(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k))) \\
&= \sum_{k=1}^{\infty} E(E(1_{F_k}(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k)) | \mathcal{P}_k)) \\
&= \sum_{k=1}^{\infty} E(1_{F_k} E(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k) | \mathcal{P}_k)) \\
&\leq \frac{1}{2\bar{\rho}} \sum_{k=1}^{\infty} \frac{E(V_Z(\tau(\mathbf{w}^k, \mathbf{Z})))}{n_k} \\
&\leq \frac{M}{2\bar{\rho}} \sum_{k=1}^{\infty} \frac{1}{n_k} < +\infty.
\end{aligned}$$

Since $D(\mathbf{w}^k) \geq 0$ a.s., $\forall k \in \mathbb{N}$, it follows from semi-martingale convergence theorem [93] that there exists $D^\infty \in L^1(\Omega)$ such that

$$D(\mathbf{w}^k) \rightarrow D^\infty \text{ a.s.}$$

As a consequence,

$$C(\mathbf{w}^k) \rightarrow C^\infty := D^\infty + R \text{ a.s.}$$

(2) From (34.6),

$$\begin{aligned}
C(\mathbf{w}^{k+1}) &\leq C(\mathbf{w}^k) - \frac{\bar{\rho}}{2} \|\mathbf{w}^k - \mathbf{w}^{k+1}\|^2 \\
&\quad + \langle \mathbf{t}^k - \nu(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle \\
&\leq C(\mathbf{w}^k) - \frac{\bar{\rho}}{2} \|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2 + \|\mathbf{t}^k - \nu(\mathbf{w}^k)\| \\
&\quad \times \|\mathbf{w}^{k+1} - \mathbf{w}^k\| \\
&\leq C(\mathbf{w}^k) + \frac{1}{\bar{\rho}} \|\mathbf{t}^k - \nu(\mathbf{w}^k)\|^2 - \frac{\bar{\rho}}{4} \|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2.
\end{aligned}$$

Therefore,

$$0 \leq \frac{\bar{\rho}}{4} \|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2 \leq C(\mathbf{w}^k) - C(\mathbf{w}^{k+1}) + \frac{1}{\bar{\rho}} \|\mathbf{t}^k - \nu(\mathbf{w}^k)\|^2.$$

On the other hand,

$$E(\|\mathbf{t}^k - v(\mathbf{w}^k)\|^2) = \frac{1}{n_k} E(V_Z(\tau(\mathbf{w}^k, \mathbf{Z}))) \leq \frac{M}{n_k}.$$

Therefore,

$$E\left(\sum_{k=1}^{\infty} \|\mathbf{t}^k - v(\mathbf{w}^k)\|^2\right) \leq M \sum_{k=1}^{\infty} \frac{1}{n_k} < +\infty.$$

It follows that

$$\sum_{k=1}^{\infty} \|\mathbf{t}^k - v(\mathbf{w}^k)\|^2 < +\infty \text{ a.s.}$$

As a consequence, $\|\mathbf{t}^k - v(\mathbf{w}^k)\| \rightarrow 0$ a.s. Therefore, we can conclude that

$$\|\mathbf{w}^{k+1} - \mathbf{w}^k\| \rightarrow 0 \text{ a.s.}$$

(3) Since $v(\mathbf{w}^k) \in \partial_w H(\mathbf{w}^k)$, we have

$$\langle v(\mathbf{w}^k), \mathbf{w}^k \rangle = H(\mathbf{w}^k) + H^*(v(\mathbf{w}^k)). \quad (34.9)$$

On the other hand, since $\mathbf{t}^k \in \partial G(\mathbf{w}^{k+1})$, we get

$$\langle \mathbf{t}^k, \mathbf{w}^{k+1} \rangle = G(\mathbf{w}^{k+1}) + G^*(\mathbf{t}^k). \quad (34.10)$$

From (34.9) and (34.10), we achieve

$$\begin{aligned} & G(\mathbf{w}^{k+1}) - H(\mathbf{w}^k) + G^*(\mathbf{t}^k) - H^*(v(\mathbf{w}^k)) \\ &= \langle \mathbf{w}^{k+1}, \mathbf{t}^k \rangle - \langle \mathbf{w}^k, v(\mathbf{w}^k) \rangle. \end{aligned} \quad (34.11)$$

By the definition of \mathbf{w}^{k+1} ,

$$G(\mathbf{w}^{k+1}) - \langle \mathbf{t}^k, \mathbf{w}^{k+1} \rangle \leq G(\mathbf{w}^k) - \langle \mathbf{t}^k, \mathbf{w}^k \rangle. \quad (34.12)$$

Combining (34.11) and (34.12), we get

$$G(\mathbf{w}^k) - H(\mathbf{w}^k) \geq H^*(v(\mathbf{w}^k)) - G^*(\mathbf{t}^k) + \langle \mathbf{w}^k, \mathbf{t}^k - v(\mathbf{w}^k) \rangle. \quad (34.13)$$

Moreover, since $v(\mathbf{w}^k) \in \partial_w H(\mathbf{w}^k)$, we have the inequality

$$H(\mathbf{w}^{k+1}) \geq H(\mathbf{w}^k) + \langle v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle. \quad (34.14)$$

Combining (34.11) and (34.14), we get

$$G(\mathbf{w}^{k+1}) - H(\mathbf{w}^{k+1}) \leq H^*(v(\mathbf{w}^k)) - G^*(\mathbf{t}^k) + \langle \mathbf{w}^{k+1}, \mathbf{t}^k - v(\mathbf{w}^k) \rangle. \quad (34.15)$$

From (34.13) and (34.15), we get the inequalities

$$\begin{aligned} C(\mathbf{w}^k) &\geq H^*(v(\mathbf{w}^k)) - G^*(\mathbf{t}^k) + \langle \mathbf{w}^k, \mathbf{t}^k - v(\mathbf{w}^k) \rangle \\ &\geq C(\mathbf{w}^{k+1}) + \langle \mathbf{w}^k - \mathbf{w}^{k+1}, \mathbf{t}^k - v(\mathbf{w}^k) \rangle. \end{aligned}$$

It is observed that $\langle \mathbf{w}^k, \mathbf{t}^k - v(\mathbf{w}^k) \rangle \rightarrow 0$ a.s and $\langle \mathbf{w}^k - \mathbf{w}^{k+1}, \mathbf{t}^k - v(\mathbf{w}^k) \rangle \rightarrow 0$ a.s, so

$$C(\mathbf{w}^k) - H^*(v(\mathbf{w}^k)) + G^*(\mathbf{t}^k) \rightarrow 0 \text{ a.s.},$$

or equivalently,

$$G(\mathbf{w}^k) - H(\mathbf{w}^k) - H^*(v(\mathbf{w}^k)) + G^*(\mathbf{t}^k) \rightarrow 0 \text{ a.s.}$$

Since $H(\mathbf{w}^k) + H^*(v(\mathbf{w}^k)) = \langle \mathbf{w}^k, v(\mathbf{w}^k) \rangle$, we get

$$G(\mathbf{w}^k) + G^*(\mathbf{t}^k) - \langle \mathbf{w}^k, v(\mathbf{w}^k) \rangle \rightarrow 0 \text{ a.s.}$$

Let \mathcal{L} be an intersection of all sets with probability 1 gained from the almost sure true statements from the beginning of the proof, $P(\mathcal{L}) = 1$ (since we have at most countably finite statements).

Let $\omega \in \mathcal{L}$; since the sequence $\{\mathbf{w}^k(\omega)\}$ is bounded, we can extract a subsequence $\{\mathbf{w}^{k_j}(\omega)\}$ such that

$$\mathbf{w}^{k_j}(\omega) \rightarrow \mathbf{w}^*(\omega) \in S.$$

By the boundedness of v and passing to subsequence if necessary, we get

$$v(\mathbf{w}^{k_j}(\omega)) \rightarrow v(\mathbf{w}^*(\omega)).$$

On the other hand,

$$\begin{aligned} \|\mathbf{t}^{k_j}(\omega) - v(\mathbf{w}^*(\omega))\| &\leq \|\mathbf{t}^{k_j}(\omega) - v(\mathbf{w}^{k_j}(\omega))\| + \|v(\mathbf{w}^{k_j}(\omega)) \\ &\quad - v(\mathbf{w}^*(\omega))\|, \end{aligned}$$

which converges to 0. Therefore,

$$\mathbf{t}^{k_j}(\omega) \rightarrow v(\mathbf{w}^*(\omega)), \quad j \rightarrow \infty.$$

We achieve the following limit:

$$\lim_{j \rightarrow \infty} (G(\mathbf{w}^{k_j}(\omega)) + G^*(\mathbf{t}^{k_j}(\omega))) = \langle \mathbf{w}^*(\omega), v(\mathbf{w}^*(\omega)) \rangle.$$

By using the semicontinuity property of

$$\theta(\mathbf{w}, \mathbf{z}) = G(\mathbf{w}) + G^*(\mathbf{z}),$$

we achieve

$$G(\mathbf{w}^*(\omega)) + G^*(v(\mathbf{w}^*(\omega))) \leq \langle \mathbf{w}^*(\omega), v(\mathbf{w}^*(\omega)) \rangle$$

and by Young inequality, the equality must hold:

$$G(\mathbf{w}^*(\omega)) + G^*(\nu(\mathbf{w}^*(\omega))) = \langle \mathbf{w}^*(\omega), \nu(\mathbf{w}^*(\omega)) \rangle.$$

In other words, $\nu(\mathbf{w}^*(\omega)) \in \partial G(\mathbf{w}^*(\omega))$. Consequently,

$$\partial G(\mathbf{w}^*(\omega)) \cap \partial H(\mathbf{w}^*(\omega)) \neq \emptyset,$$

and we conclude that $\mathbf{w}^*(\omega)$ is a critical point of $C = G - H$. The proof is completed.

34.4.3 Online Stochastic DCA2

The whole set of assumptions for online stochastic DCA1 is applied to online stochastic DCA2, together with the following assumption.

Assumption 4

(1) *There exists a Borel measurable function*

$$\tilde{g} : \mathbb{R}^n \rightarrow \mathbb{R}$$

such that $\int_{\Xi} |\tilde{g}(z)| dP_Z < +\infty$ and

$$\text{for all } \mathbf{w} \in S, z \in \Xi, \quad |g(\mathbf{w}, z)| \leq \tilde{g}(z).$$

(2) *For all $\mathbf{w} \in S, E(g(\mathbf{w}, \mathbf{Z})^2) < +\infty$.*

(3) *The family of functions $\mathcal{G} = \{g(\cdot, z) | z \in \Xi\}$ is equicontinuous in S , i.e., for each $\mathbf{w} \in S$, for every $\epsilon > 0$, there exists $\delta > 0$ such that*

$$|g(\mathbf{w}, z) - g(\mathbf{w}', z)| < \epsilon, \forall z \in \Xi, \forall \mathbf{w}, \mathbf{w}' \in S : \|\mathbf{w} - \mathbf{w}'\| < \delta.$$

(4) *For all $\mathbf{w} \in S, g(\mathbf{w}, \cdot)$ is lower semicontinuous.*

Notice that, due to the compactness of S , the equicontinuity of \mathcal{G} is equivalent to \mathcal{G} being uniform equicontinuous, i.e., for every $\epsilon > 0$, there exists $\delta > 0$ such that

$$|g(\mathbf{w}, z) - g(\mathbf{w}', z)| < \epsilon, \forall z \in \Xi, \forall \mathbf{w}, \mathbf{w}' \in S : \|\mathbf{w} - \mathbf{w}'\| < \delta.$$

Let $\{\mathbf{Z}_p\}_{p=1}^\infty$ be a sequence of independent random variables which is of identical distribution with \mathbf{Z} . We prove that the family

$$\mathcal{J} = \left\{ \sup_{\mathbf{w} \in S} \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i) - E(g(\mathbf{w}, \mathbf{Z})) \right| : p \in \mathbb{N} \right\}$$

is uniform integrable.

Proof. The proof is divided into two parts.

(a) We show that there is $\Upsilon > 0$ such that for all $X \in \mathcal{J}$, $E(|X|) \leq \Upsilon$.

Thus, due to the uniform equicontinuity of \mathcal{G} , there is $\delta > 0$ such that

$$\forall z \in \Xi, \mathbf{w}, \mathbf{w}' \in S : \|\mathbf{w} - \mathbf{w}'\| < \delta, |g(\mathbf{w}, z) - g(\mathbf{w}', z)| < 1.$$

Since $S \subset \cup_{\mathbf{w} \in S} B(\mathbf{w}, \delta)$ and S is compact, there exists $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r \in S$ such that

$$S \subset \cup_{i=1}^r B(\mathbf{w}_i, \delta).$$

Let $\mathbf{w} \in S$; there is $j \in \{1, 2, \dots, r\}$ such that

$$\mathbf{w} \in B(\mathbf{w}_j, \delta).$$

It follows that

$$|g(\mathbf{w}, z) - g(\mathbf{w}_j, z)| < 1, \forall z \in \Xi.$$

Let $p \in \mathbb{N}$; on $\cap_{k=1}^\infty (\mathbf{Z}_k \in \Xi)$, we have

$$\begin{aligned} & \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i) - E(g(\mathbf{w}, \mathbf{Z})) \right| \\ & \leq \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right| \\ & \quad + p^{-1} \sum_{i=1}^p |g(\mathbf{w}, \mathbf{Z}_i) - g(\mathbf{w}_j, \mathbf{Z}_i)| + E(|g(\mathbf{w}, \mathbf{Z}) - g(\mathbf{w}_j, \mathbf{Z})|) \\ & \leq \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right| + 2. \end{aligned}$$

As a consequence, for all $\mathbf{w} \in S$,

$$\begin{aligned} & \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i) - E(g(\mathbf{w}, \mathbf{Z})) \right| \leq \max_{j=1, \dots, r} \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right| + 2. \end{aligned}$$

It follows that

$$\begin{aligned} & \sup_{\mathbf{w} \in S} \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i) - E(g(\mathbf{w}, \mathbf{Z})) \right| \leq \max_{j=1, \dots, r} \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right| + 2. \end{aligned}$$

On the other hand,

$$\begin{aligned} \max_{j=1,r} |p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z}))| &\leq \sum_{j=1}^r \\ &\leq \sqrt{P(A)} \cdot \sum_{j=1}^r V(g(\mathbf{w}_j, \mathbf{Z}))^{\frac{1}{2}} + 2P(A) \\ |p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z}))| &\leq \Upsilon \sqrt{P(A)}, \end{aligned}$$

By the independence of \mathbf{Z}_i 's,

$$\begin{aligned} &E \left(\left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right| \right) \\ &\leq E \left(\left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right|^2 \right)^{\frac{1}{2}} \\ &= \frac{V(g(\mathbf{w}_j, \mathbf{Z}))^{\frac{1}{2}}}{\sqrt{p}} \leq V(g(\mathbf{w}_j, \mathbf{Z}))^{\frac{1}{2}}. \end{aligned}$$

Therefore,

$$\begin{aligned} &E \left(\sup_{\mathbf{w} \in S} \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i) - E(g(\mathbf{w}, \mathbf{Z})) \right| \right) \\ &\leq \sum_{j=1}^r V(g(\mathbf{w}_j, \mathbf{Z}))^{\frac{1}{2}} + 2 := \Upsilon, \end{aligned}$$

where Υ is finite due to the finiteness of the second moment of $g(\mathbf{w}, \mathbf{Z})$ (Assumption 4(2)).

- (b) We next prove that for every $\epsilon > 0$, there exists $\eta > 0$ such that, for every measurable set A such that $P(A) \leq \eta$ and every $X \in \mathcal{J}$, $E(|X| \mathbf{1}_A) \leq \epsilon$.

Thus, let $X \in \mathcal{J}$, from (a)

$$|X| \leq \sum_{j=1}^r \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right| + 2.$$

Therefore,

$$\begin{aligned} E(|X| \mathbf{1}_A) &\leq 2P(A) + \sum_{j=1}^r \\ &E \left(\left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right| \mathbf{1}_A \right) \\ &\leq 2P(A) + \sqrt{P(A)} \sum_{j=1}^r \\ &E \left(\left| p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i) - E(g(\mathbf{w}_j, \mathbf{Z})) \right|^2 \right)^{\frac{1}{2}} \end{aligned}$$

which can be arbitrarily small.

We conclude that \mathcal{J} is uniform integrable. \square

Moreover, the uniform strong law of large number ensures that

$$\sup_{\mathbf{w} \in S} \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i) - E(g(\mathbf{w}, \mathbf{Z})) \right| \rightarrow 0 \text{ a.s.} \quad (34.16)$$

The uniform strong law of large numbers (34.16) can be found in [94], but here we give a detail proof in appendix B.1 for the sack of completeness.

It follows from Vitali convergence theorem [95] that

$$E \left(\sup_{\mathbf{w} \in S} \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i) - E(g(\mathbf{w}, \mathbf{Z})) \right| \right) \rightarrow 0$$

when $p \rightarrow \infty$.

Now let $\{\alpha_k\}$ be a sequence of positive number such that $\sum_{k=1}^{\infty} \alpha_k < \infty$; we can construct a sequence $\{N_k\}$ satisfying

$$\forall p \geq N_k, E \left(\sup_{\mathbf{w} \in S} \left| p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i) - E(g(\mathbf{w}, \mathbf{Z})) \right| \right) < \alpha_k. \quad (34.17)$$

We make an additional setting before giving a new online stochastic DCA scheme.

For every $r \in \mathbb{N}$, we consider a function called Θ_r :

$$\begin{aligned} \Theta_r &: \mathbb{R}^{nr} \times \mathbb{R}^{2m} \rightarrow \mathbb{R} \cup \{+\infty\} \\ (\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_r, \mathbf{t}, \mathbf{w}) &\mapsto \frac{1}{r} \sum_{i=1}^r g(\mathbf{w}, \mathbf{z}_i) - \langle \mathbf{t}, \mathbf{w} \rangle. \end{aligned}$$

Under Assumptions 4(3),(4), g is lower semicontinuous; therefore, Θ_r is lower semicontinuous.

Furthermore, for each $(\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_r) \in \Xi^r$ and for $\mathbf{t} \in \mathbb{R}^m$,

$$\arg \min_{\mathbf{w} \in \mathbb{R}^m} \Theta_r(\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_r, \mathbf{t}, \mathbf{w}) \neq \emptyset.$$

By selection theorem, there exists a Borel measurable function $\Gamma_r : \mathbb{R}^{nr} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$ such that

$$\Gamma_r(\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_r, \mathbf{t}) \in \arg \min_{\mathbf{w} \in \mathbb{R}^m} \Theta_r(\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_r, \mathbf{t}, \mathbf{w})$$

for all $z_1, z_2, \dots, z_r \in \Xi$ and $t \in \mathbb{R}^m$. We now propose a new DCA scheme as the following.

Algorithm 2 Online stochastic DCA2.

Step 1. Choose $w^0 \in S$.

Step 2. For $k = 0, 1, 2, \dots$ do

1. Draw independently $Z_{k,1}, Z_{k,2}, \dots, Z_{k,n_k}$ from Z in such a way they are also independent to the past.
2. Compute $t^k = \frac{1}{n_k} \sum_{i=1}^{n_k} \tau(w^k, Z_{k,i})$.
3. Solve the convex program to get w^{k+1}

$$w^{k+1} = \Gamma_{n_k}(Z_{k,1}, Z_{k,2}, \dots, Z_{k,n_k}, t^k) \in \arg \min_{w \in \mathbb{R}^m} \left\{ \frac{1}{n_k} \sum_{i=1}^{n_k} g(w, Z_{k,i}) - \langle t^k, w \rangle \right\}.$$

4. If stopping criteria are satisfied, break. □

The algorithm is well-defined with probability 1. We study convergence properties of online stochastic DCA2 in the set of events which makes the algorithm works.

To be more specific, the set of events that make the algorithm works is

$$\mathcal{V} = \bigcap_{k=1}^{\infty} \bigcap_{i=1}^{n_k} (Z_{k,i} \in \Xi).$$

By the Borel measurability of Γ_r and τ , it follows by induction that w^k is \mathcal{P}_k measurable, for all $k \geq 1$, and t^k is \mathcal{P}_{k+1} measurable, for all $k \geq 0$.

Theorem 34.8 Let $0 < \bar{\rho} < \rho(H)$. Under Assumptions 1, 2, 3, 4 (Assumption 3 applying to the sequence $\{w^k\}$ generated by online stochastic DCA2), if the sequence of sample size satisfies

$$\sum_{k=1}^{\infty} \frac{1}{n_k} < \infty \text{ and } n_k \geq N_k, \forall k \in \mathbb{N},$$

where $\{N_k\}$ is determined by (34.17), the iterates of online stochastic DCA2 satisfy:

(1) There exists $C^\infty \in L^1(\Omega)$ such that

$$C(w^k) \rightarrow C^\infty \text{ a.s.}$$

(2) $\|w^{k+1} - w^k\| \rightarrow 0$ a.s.

(3) There exists a measurable set \mathcal{Y} with $P(\mathcal{Y}) = 1$ such that for every $\omega \in \mathcal{Y}$, the sequence $\{w^k(\omega)\}$ has a subsequence converging to $w^*(\omega)$ which is a critical point of $C = G - H$.

Proof. (1) Since $v(w^k) \in \partial H(w^k)$, we get

$$H(w^{k+1}) \geq H(w^k) + \langle v(w^k), w^{k+1} - w^k \rangle + \frac{\bar{\rho}}{2} \|w^{k+1} - w^k\|^2. \tag{34.18}$$

By the definition of w^{k+1} ,

$$\frac{1}{n_k} \sum_{i=1}^{n_k} g(w^{k+1}, Z_{k,i}) - \langle t^k, w^{k+1} \rangle \leq \frac{1}{n_k} \sum_{i=1}^{n_k} g(w^k, Z_{k,i}) - \langle t^k, w^k \rangle. \tag{34.19}$$

Combining these two inequalities, we get

$$\begin{aligned} & \frac{1}{n_k} \sum_{i=1}^{n_k} g(w^{k+1}, Z_{k,i}) - H(w^{k+1}) \\ & \leq \frac{1}{n_k} \sum_{i=1}^{n_k} g(w^k, Z_{k,i}) - \frac{\bar{\rho}}{2} \|w^{k+1} - w^k\|^2 \\ & \quad + \langle t^k - v(w^k), w^{k+1} - w^k \rangle - H(w^k). \end{aligned} \tag{34.20}$$

By taking the conditional expectation (34.20) with respect to \mathcal{P}_k and noticing that

$$E \left(\frac{1}{n_k} \sum_{i=1}^{n_k} g(w^k, Z_{k,i}) | \mathcal{P}_k \right) = G(w^k),$$

$$E(\langle t^k - v(w^k), w^{k+1} - w^k \rangle | \mathcal{P}_k) \leq E(\|w^{k+1} - w^k\|^2 | \mathcal{P}_k)^{\frac{1}{2}} \frac{V_Z(\tau(w^k, Z))^{1/2}}{\sqrt{n_k}},$$

we get

$$\begin{aligned} & E(C(w^{k+1}) - C(w^k) | \mathcal{P}_k) \\ & \leq -\frac{\bar{\rho}}{2} E(\|w^{k+1} - w^k\|^2 | \mathcal{P}_k) \\ & \quad + E \left(G(w^{k+1}) - \frac{1}{n_k} \sum_{i=1}^{n_k} g(w^{k+1}, Z_{k,i}) | \mathcal{P}_k \right) \\ & \quad + E(\|w^{k+1} - w^k\|^2 | \mathcal{P}_k)^{\frac{1}{2}} \frac{V_Z(\tau(w^k, Z))^{1/2}}{\sqrt{n_k}} \\ & \leq E \left(\sup_{w \in S} \left| G(w) - \frac{1}{n_k} \sum_{i=1}^{n_k} g(w, Z_{k,i}) \right| | \mathcal{P}_k \right) \\ & \quad + \frac{V_Z(\tau(w^k, Z))}{2\bar{\rho} \times n_k}. \end{aligned}$$

By Assumption 2(3), there exists R such that

$$C(w) \geq R, \forall w \in S.$$

Let $D(\mathbf{w}) = C(\mathbf{w}) - R$; we get

$$E(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k) | \mathcal{P}_k)$$

$$\leq \frac{V_{\mathbf{Z}}(\tau(\mathbf{w}^k, \mathbf{Z}))}{2\bar{\rho} \times n_k} + E \left(\sup_{\mathbf{w} \in S} | G(\mathbf{w}) - \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}) | \mid \mathcal{P}_k \right).$$

Let $F_n = [E(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k) | \mathcal{P}_k) > 0]$,

$$\begin{aligned} & \sum_{k=1}^{\infty} E(1_{F_k}(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k))) \\ &= \sum_{k=1}^{\infty} E(E(1_{F_k}(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k)) | \mathcal{P}_k)) \\ &= \sum_{k=1}^{\infty} E(1_{F_k} E(D(\mathbf{w}^{k+1}) - D(\mathbf{w}^k) | \mathcal{P}_k)) \\ &\leq \sum_{k=1}^{\infty} E \left(\sup_{\mathbf{w} \in S} | G(\mathbf{w}) - \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}) | \right) \\ &\quad + \frac{M}{2\bar{\rho}} \sum_{k=1}^{\infty} \frac{1}{n_k} \\ &\leq \sum_{k=1}^{\infty} \alpha_k + \frac{M}{2\bar{\rho}} \sum_{k=1}^{\infty} \frac{1}{n_k} < \infty. \end{aligned}$$

By applying semimartingale convergence theorem, there exists $D^\infty \in L^1(\Omega)$ such that

$$D(\mathbf{w}^k) \rightarrow D^\infty \text{ a.s.}$$

As a consequence, $C^\infty = D^\infty + R \in L^1(\Omega)$ and

$$C(\mathbf{w}_k) \rightarrow C^\infty \text{ a.s.}$$

(2) From (34.20), using the AM-GM inequality

$$\begin{aligned} \langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle &\leq \frac{1}{\bar{\rho}} \|\mathbf{t}^k - v(\mathbf{w}^k)\|^2 \\ &\quad + \frac{\bar{\rho}}{4} \|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2, \end{aligned}$$

we achieve

$$\begin{aligned} \frac{\bar{\rho}}{4} \|\mathbf{w}^{k+1} - \mathbf{w}^k\|^2 &\leq \frac{1}{\bar{\rho}} \|\mathbf{t}^k - v(\mathbf{w}^k)\|^2 + H(\mathbf{w}^{k+1}) - H(\mathbf{w}^k) \\ &\quad + \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}^k, \mathbf{Z}_{k,i}) - \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}^{k+1}, \mathbf{Z}_{k,i}) \\ &= \frac{1}{\bar{\rho}} \|\mathbf{t}^k - v(\mathbf{w}^k)\|^2 - C(\mathbf{w}^{k+1}) + C(\mathbf{w}^k) \\ &\quad + G(\mathbf{w}^{k+1}) + \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}^k, \mathbf{Z}_{k,i}) - G(\mathbf{w}^k) \end{aligned}$$

$$- \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}^{k+1}, \mathbf{Z}_{k,i}).$$

By uniform strong law of large number,

$$\|\mathbf{t}^k - v(\mathbf{w}^k)\| \rightarrow 0 \text{ a.s.},$$

$$\frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}^k, \mathbf{Z}_{k,i}) - G(\mathbf{w}^k) \rightarrow 0 \text{ a.s.},$$

$$G(\mathbf{w}^{k+1}) - \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}^{k+1}, \mathbf{Z}_{k,i}) \rightarrow 0 \text{ a.s.},$$

we conclude that $\|\mathbf{w}^{k+1} - \mathbf{w}^k\| \rightarrow 0$ a.s.

The uniform strong law of large number for triangular arrays,

$$\sup_{\mathbf{w} \in S} | \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}) - G(\mathbf{w}) | \rightarrow 0 \text{ a.s.} \quad (34.21)$$

is proven in appendix B.2.

(3) Since $v(\mathbf{w}^k) \in \partial_{\mathbf{w}} H(\mathbf{w}^k)$, we have

$$\langle v(\mathbf{w}^k), \mathbf{w}^k \rangle = H(\mathbf{w}^k) + H^*(v(\mathbf{w}^k)). \quad (34.22)$$

Denoting $G_k(\mathbf{w}) = \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i})$, $\forall \mathbf{w} \in \mathbb{R}^m$, it follows from the definition of \mathbf{w}^{k+1} that

$$\langle \mathbf{w}^{k+1}, \mathbf{t}^k \rangle = G_k(\mathbf{w}^{k+1}) + G_k^*(\mathbf{t}^k). \quad (34.23)$$

From (34.22) and (34.23),

$$\begin{aligned} G_k(\mathbf{w}^{k+1}) - H(\mathbf{w}^k) + G_k^*(\mathbf{t}^k) - H^*(v(\mathbf{w}^k)) \\ = \langle \mathbf{w}^{k+1}, \mathbf{t}^k \rangle - \langle v(\mathbf{w}^k), \mathbf{w}^k \rangle. \end{aligned} \quad (34.24)$$

Combine (34.24) with the following inequality

$$H(\mathbf{w}^{k+1}) \geq H(\mathbf{w}^k) + \langle v(\mathbf{w}^k), \mathbf{w}^{k+1} - \mathbf{w}^k \rangle,$$

to get

$$\begin{aligned} G_k(\mathbf{w}^{k+1}) - H(\mathbf{w}^{k+1}) + G_k^*(\mathbf{t}^k) - H^*(v(\mathbf{w}^k)) \\ \leq \langle \mathbf{w}^{k+1}, \mathbf{t}^k - v(\mathbf{w}^k) \rangle. \end{aligned} \quad (34.25)$$

On the other hand, by combining (34.24) with the following inequality

$$G_k(\mathbf{w}^{k+1}) - \langle \mathbf{t}^k, \mathbf{w}^{k+1} \rangle \leq G_k(\mathbf{w}^k) - \langle \mathbf{t}^k, \mathbf{w}^k \rangle$$

we get

$$G_k(\mathbf{w}^k) - H(\mathbf{w}^k) + G_k^*(\mathbf{t}^k) - H^*(v(\mathbf{w}^k)) \geq \langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^k \rangle. \quad (34.26)$$

From (34.25) and (34.26), we have these inequalities

$$\begin{aligned} G_k(\mathbf{w}^k) - H(\mathbf{w}^k) &\geq H^*(v(\mathbf{w}^k)) - G_k^*(\mathbf{t}^k) + \langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^k \rangle \\ &\geq G_k(\mathbf{w}^{k+1}) - H(\mathbf{w}^{k+1}) \\ &\quad + \langle \mathbf{t}^k - v(\mathbf{w}^k), \mathbf{w}^k - \mathbf{w}^{k+1} \rangle. \end{aligned}$$

By uniform strong law of large number,

$$\begin{aligned} G_k(\mathbf{w}^k) - H(\mathbf{w}^k) &\rightarrow C^\infty \text{ a.s.}, \\ G_k(\mathbf{w}^{k+1}) - H(\mathbf{w}^{k+1}) &\rightarrow C^\infty \text{ a.s.}, \\ \|\mathbf{t}^k - v(\mathbf{w}^k)\| &\rightarrow 0 \text{ a.s.}, \end{aligned}$$

we are left with

$$G_k(\mathbf{w}^k) + G_k^*(\mathbf{t}^k) - H(\mathbf{w}^k) - H^*(v(\mathbf{w}^k)) \rightarrow 0 \text{ a.s.},$$

or equivalently,

$$G(\mathbf{w}^k) + G_k^*(\mathbf{t}^k) - \langle \mathbf{w}^k, v(\mathbf{w}^k) \rangle \rightarrow 0 \text{ a.s.}$$

We have an observation that

$$\begin{aligned} |G_k^*(\mathbf{t}^k) - G^*(\mathbf{t}^k)| &= \left| \sup_{\mathbf{x} \in S} \{ \langle \mathbf{x}, \mathbf{t}^k \rangle - G_k(\mathbf{x}) \} - \sup_{\mathbf{x} \in S} \{ \langle \mathbf{x}, \mathbf{t}^k \rangle - G(\mathbf{x}) \} \right| \\ &\leq \sup_{\mathbf{x} \in S} |G_k(\mathbf{x}) - G(\mathbf{x})| \rightarrow 0 \text{ a.s.}, \end{aligned}$$

here we use the inequality

$$\left| \sup_{\mathbf{x} \in S} u(\mathbf{x}) - \sup_{\mathbf{x} \in S} v(\mathbf{x}) \right| \leq \sup_{\mathbf{x} \in S} |u(\mathbf{x}) - v(\mathbf{x})|.$$

By this evaluation we get the result

$$G(\mathbf{w}^k) + G^*(\mathbf{t}^k) - \langle \mathbf{w}^k, v(\mathbf{w}^k) \rangle \rightarrow 0 \text{ a.s.}$$

With the same argument as the proof of online stochastic DCA1, we conclude that the sequence $\{\mathbf{w}^k\}$ admits a subsequence converging to a critical point of $C = G - H$ with probability 1.

34.5 Conclusion

In this chapter, we have studied ways to combine stochastic optimization in an online manner and DCA to make use of their premier features. The main idea is quite simple; some deterministic quantities in the traditional DCA are replaced by their corresponding stochastic approximations in order to reduce the computational cost or deal with uncertainty. Intuitively, the more deterministic quantities we replace, the

more instability the new algorithm gets. In the first online stochastic algorithm, we only use stochastic approximation for the subgradient component, while in the second one, the replacement is applied for two components including the subgradient and G . The first scheme admits an explicit condition for the rate of sample size, while an implicit condition appears in the second algorithm. For both algorithms, the number of samples used increases over time to make the approximations more accurate. Consequently, these two online stochastic algorithms are similar to the traditional DCA as time step tends to infinity. This leads to similar behaviors between online stochastic DCA and traditional DCA. Thus, these two schemes have convergence properties in terms of subsequence to critical points with probability 1. Besides the wide class of one-stage stochastic programming problems mentioned in the chapter, multistage stochastic programming is such a promising area where stochastic optimization will find many deterministic quantities to substitute. Furthermore, a deterministic function sometimes has its convenient probabilistic representation, so it would be a chance for stochastic algorithms to prove their capability. Regarding DC programming and DCA, since its day of birth in 1985, its various variations have been studied extensively such as proximal DCA [96], DCA with successive DC decomposition [1], etc. Therefore, it is a natural idea that each of these variations should have corresponding stochastic versions to offer us more options in practice as well as many nice results interfacing between probability and optimization theory to discover.

34.6 Appendix A: A Theorem on Conditional Expectation

This appendix states and proves a theorem which is used frequently in our study to manipulate conditional expectation.

Theorem A.1 *Let (Ω, \mathcal{M}, P) be a probability space and let \mathcal{P} be a sub-sigma-algebra of \mathcal{M} .*

Let \mathbf{X} be a n -dimensional random vector which is \mathcal{P} -measurable and let \mathbf{Y} be a m -dimensional random vector which is \mathcal{P} -independent.

Let $\alpha : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^d$ be Borel measurable such that $E(\|\alpha(\mathbf{X}, \mathbf{Y})\|) < +\infty$. Then the following equality holds,

$$E(\alpha(\mathbf{X}, \mathbf{Y})|\mathcal{P}) = E_Y(\alpha(\mathbf{X}, \mathbf{Y})).$$

Proof. Let μ_X, μ_Y be probability measures on \mathbb{R}^n and \mathbb{R}^m generated by \mathbf{X} and \mathbf{Y} , respectively.

Since

$$\int_{\mathbb{R}^n \times \mathbb{R}^m} \|\alpha(\mathbf{x}, \mathbf{y})\| d(\mu_X \times \mu_Y) = E(\|\alpha(\mathbf{X}, \mathbf{Y})\|)$$

is finite, we have $\int_{\mathbb{R}^n} \|\alpha(\mathbf{x}, \mathbf{y})\| d(\mu_Y^*) < \infty$, for μ_X almost sure. Equivalently,

$$g(\mathbf{x}) := E(\alpha(\mathbf{x}, \mathbf{Y}))$$

is well-defined for μ_X almost sure. Moreover, we have $g \in L^1(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n), \mu_X)$.

For every set $A \in \mathcal{P}$, we show that

$$E(\alpha(\mathbf{X}, \mathbf{Y})1_A) = E(g(\mathbf{X})1_A). \quad (\text{A.27})$$

If $P(A) = 0$, (A.27) is obviously true.

Consider $P(A) > 0$, (A.27) can be written as

$$\int_A \alpha(\mathbf{X}, \mathbf{Y}) dP = \int_A g(\mathbf{X}) dP. \quad (\text{A.28})$$

Let (A, \mathcal{M}_A, P_A) be a probability space induced by (Ω, \mathcal{M}, P) on A , then (A.28) can be rewritten as

$$\int_A \alpha(\mathbf{X}, \mathbf{Y}) dP_A = \int_A g(\mathbf{X}) dP_A.$$

It is observed that,

$$\int_A \alpha(\mathbf{X}, \mathbf{Y}) dP_A = \int_{\mathbb{R}^{m+n}} \alpha(\mathbf{x}, \mathbf{y}) d\pi^*,$$

where π^* is a probability measure on \mathbb{R}^{m+n} defined by

$$\pi^*(D) = P_A(A \cap [(X, Y) \in D]), \forall D \in \mathcal{B}(\mathbb{R}^{m+n}).$$

Let μ_X^*, μ_Y^* be probability measures on \mathbb{R}^n and \mathbb{R}^m corresponding to X and Y , where X, Y are viewed as random vectors defined on probability space A ,

$$\begin{aligned} \mu_X^*(B) &= P_A(A \cap [X \in B]), \quad \forall B \in \mathcal{B}(\mathbb{R}^n), \\ \mu_Y^*(C) &= P_A(A \cap [Y \in C]), \quad \forall C \in \mathcal{B}(\mathbb{R}^m). \end{aligned}$$

We show that $\pi^* = \mu_X^* \times \mu_Y^*$.

Thus, let $B \in \mathcal{B}(\mathbb{R}^n)$, $C \in \mathcal{B}(\mathbb{R}^m)$.

$$\begin{aligned} \pi^*(B \times C) &= P_A(A \cap [(X, Y) \in B \times C]) \\ &= P_A(A \cap [X \in B] \cap [Y \in C]) \\ &= \frac{P(A \cap [X \in B] \cap [Y \in C])}{P(A)} \\ &= \frac{P(A \cap [X \in B])P(Y \in C)}{P(A)}. \end{aligned}$$

since $A \cap [X \in B]$ and $[Y \in C]$ are independent.

On the other hand,

$$\begin{aligned} (\mu_X^* \times \mu_Y^*)(B \times C) &= \mu_X^*(B)\mu_Y^*(C) \\ &= P_A(A \cap [X \in B]) \times P_A(A \cap [Y \in C]) \\ &= \frac{P(A \cap [X \in B])}{P(A)} \times \frac{P(A \cap [Y \in C])}{P(A)} \\ &= \frac{P(A \cap [X \in B])P(Y \in C)}{P(A)}, \end{aligned}$$

since A and $[Y \in C]$ are independent.

Consequently, $\pi^* \equiv \mu_X^* \times \mu_Y^*$ due to the uniqueness of product measure.

Therefore,

$$\int_{\mathbb{R}^{m+n}} \alpha(\mathbf{x}, \mathbf{y}) d\pi^* = \int_{\mathbb{R}^{m+n}} \alpha(\mathbf{x}, \mathbf{y}) d(\mu_X^* \times \mu_Y^*).$$

By applying Fubini theorem, we get

$$\int_{\mathbb{R}^{m+n}} \alpha(\mathbf{x}, \mathbf{y}) d(\mu_X^* \times \mu_Y^*) = \int_{\mathbb{R}^n} \int_{\mathbb{R}^m} \alpha(\mathbf{x}, \mathbf{y}) d\mu_Y^* d\mu_X^*$$

We show that $\mu_Y = \mu_Y^*$. Thus, let $C \in \mathcal{B}(\mathbb{R}^m)$, we have

$$\begin{aligned} \mu_Y^*(C) &= P_A(A \cap [Y \in C]) \\ &= \frac{P(A \cap [Y \in C])}{P(A)} \\ &= P(Y \in C) = \mu_Y(C). \end{aligned}$$

Therefore, we achieve

$$\begin{aligned} \int_{\mathbb{R}^{m+n}} \alpha(\mathbf{x}, \mathbf{y}) d(\mu_X^* \times \mu_Y^*) &= \int_{\mathbb{R}^n} \int_{\mathbb{R}^m} \alpha(\mathbf{x}, \mathbf{y}) d\mu_Y d\mu_X^* \\ &= \int_{\mathbb{R}^n} \int_{\Omega} \alpha(\mathbf{x}, \mathbf{Y}) dP d\mu_X^* \\ &= \int_{\mathbb{R}^n} g(\mathbf{x}) d\mu_X^* = \int_A g(\mathbf{X}) dP_A, \end{aligned}$$

the proof is completed.

Remark Roughly speaking, the theorem says that, if the information \mathcal{P} is known, it follows that X is known (since X is \mathcal{P} -measurable) and then X is fixed, denoted by $X\mathcal{P}$. On the other hand, \mathcal{P} provides no information about Y . Consequently, the conditional expectation becomes the expectation in the normal sense, $E(\alpha(X\mathcal{P}, Y))$.

34.7 Appendix B: Uniform Strong Law of Large Number

This appendix proves two uniform strong laws of large number that are used for studying online stochastic DCA.

B.1 Uniform Strong Law of Large Number for a Sequence

In this section, we prove (34.16).

Let $\epsilon \in \mathbb{Q}^+$, there exists $\delta > 0$ such that

$$\forall z \in \mathcal{E}, \mathbf{w}, \mathbf{w}' \in S : \|\mathbf{w} - \mathbf{w}'\| < \delta, |g(\mathbf{w}, z) - g(\mathbf{w}', z)| < \epsilon.$$

By the axiom of choice, we fix the way of choosing δ , denoted by δ_ϵ , i.e, δ can be seen as a function of ϵ . This idea is helpful at the end of the proof.

Since $S \subset \cup_{\mathbf{w} \in S} B(\mathbf{w}, \delta_\epsilon)$ and S is compact, there are $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r \in S$ such that

$$S \subset \cup_{i=1}^r B(\mathbf{w}_i, \delta_\epsilon).$$

Again, we fix the way of choosing one set $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r\}$, denoted by $\mathcal{K}(\epsilon) = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r\}$.

Let $\Lambda := \cap_{k=1}^\infty (\mathbf{Z}_k \in \mathcal{E})$, then $P(\Lambda) = 1$.

Let $\omega \in \Lambda$, $p \in \mathbb{N}$ and $\mathbf{w} \in S$, there exists j in $\{1, 2, \dots, r\}$ such that $\mathbf{w} \in B(\mathbf{w}_j, r)$, it follows that

$$\begin{aligned} |p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}, \mathbf{Z}))| &\leq |p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i(\omega)) \\ &\quad - E(g(\mathbf{w}_j, \mathbf{Z}))| + 2\epsilon. \end{aligned}$$

Therefore

$$\begin{aligned} &|p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}, \mathbf{Z}))| \\ &\leq \max_{j=1, r} |p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z}))| + 2\epsilon. \end{aligned}$$

As a consequence

$$\begin{aligned} &\sup_{\mathbf{w} \in S} |p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}, \mathbf{Z}))| \\ &\leq \max_{j=1, r} |p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z}))| + 2\epsilon. \quad (\text{B.1}) \end{aligned}$$

Let $\Lambda_\epsilon \subset \Lambda$ be a set of event ω satisfying

$$\max_{j=1, r} |p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z}))| \rightarrow 0.$$

Here we write Λ_ϵ to emphasize the dependence of $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r\}$ to ϵ . By strong law of large number, it follows that $P(\Lambda_\epsilon) = 1$. Let

$$\bar{\Lambda} = \cap_{\epsilon \in \mathbb{Q}^+} \Lambda_\epsilon,$$

then $P(\bar{\Lambda}) = 1$. We now prove

$$\forall \omega \in \bar{\Lambda}, \sup_{\mathbf{w} \in S} |p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}, \mathbf{Z}))| \rightarrow 0.$$

Let $\eta > 0$, we choose $\epsilon \in \mathbb{Q}^+$ and $\epsilon < \frac{\eta}{3}$. We can recover the inequality (B.1) because the choice mappings are fixed from the beginning. On the other hand, there exists M such that $\forall p \geq M$,

$$\max_{j=1, r} |p^{-1} \sum_{i=1}^p g(\mathbf{w}_j, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z}))| < \frac{\eta}{3}.$$

Therefore, for all $p \geq M$,

$$\sup_{\mathbf{w} \in S} |p^{-1} \sum_{i=1}^p g(\mathbf{w}, \mathbf{Z}_i(\omega)) - E(g(\mathbf{w}, \mathbf{Z}))| < \eta.$$

The proof is completed.

B.2 Uniform Strong Law of Large Number for Triangular Arrays

In this section, we prove (34.21),

$$\sup_{\mathbf{w} \in S} | \frac{1}{n_k} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}) - E(g(\mathbf{w}, \mathbf{Z})) | \rightarrow 0 \text{ a.s.}$$

Let $\epsilon \in \mathbb{Q}^+$, there exists $\delta > 0$ such that

$$\forall z \in \mathcal{E}, \mathbf{w}, \mathbf{w}' \in S : \|\mathbf{w} - \mathbf{w}'\| < \delta, |g(\mathbf{w}, z) - g(\mathbf{w}', z)| < \epsilon.$$

We fix the way of choosing δ , denoted by δ_ϵ .

Since $S \subset \cup_{\mathbf{w} \in S} B(\mathbf{w}, \delta_\epsilon)$ and S is compact, there are $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r \in S$ such that

$$S \subset \cup_{i=1}^r B(\mathbf{w}_i, \delta_\epsilon).$$

We also fix the way of choosing the set $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r\}$, denoted by $\mathcal{K}(\epsilon) = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r\}$.

Let $\Lambda := \cap_{k=1}^\infty \cap_{i=1}^{n_k} (\mathbf{Z}_{k,i} \in \mathcal{E})$, then $P(\Lambda) = 1$.

Let $\omega \in \Lambda$, $k \in \mathbb{N}$ and $\mathbf{w} \in S$, there exists j in $\{1, 2, \dots, r\}$ such that $\mathbf{w} \in B(\mathbf{w}_j, r)$, it follows that

$$\begin{aligned} &|n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}, \mathbf{Z}))| \leq |n_k^{-1} \sum_{i=1}^{n_k} \\ &g(\mathbf{w}_j, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z}))| + 2\epsilon. \end{aligned}$$

Therefore

$$\begin{aligned} & \left| n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}, \mathbf{Z})) \right| \\ & \leq \max_{j=1,r} | n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}_j, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z})) | + 2\epsilon. \end{aligned}$$

As a consequence,

$$\begin{aligned} & \sup_{\mathbf{w} \in S} | n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}, \mathbf{Z})) | \\ & \leq \max_{j=1,r} | n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}_j, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z})) | + 2\epsilon. \end{aligned} \quad (\text{B.2})$$

Let $\Lambda_\epsilon \subset \Lambda$ be a set of event ω satisfying

$$\max_{j=1,r} | n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}_j, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z})) | \rightarrow 0.$$

We show that $P(\Lambda_\epsilon) = 1$. Thus, for each j in $\{1, 2, \dots, r\}$,

$$E \left(\left| n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}_j, \mathbf{Z}_{k,i}) - E(g(\mathbf{w}_j, \mathbf{Z})) \right|^2 \right) \leq \frac{V(g(\mathbf{w}_j, \mathbf{Z}))}{n_k}.$$

Therefore

$$\begin{aligned} & E \left(\sum_{k=1}^{\infty} \left| n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}_j, \mathbf{Z}_{k,i}) - E(g(\mathbf{w}_j, \mathbf{Z})) \right|^2 \right) \\ & \leq V(g(\mathbf{w}_j, \mathbf{Z})) \sum_{k=1}^{\infty} \frac{1}{n_k} < \infty. \end{aligned}$$

It follows that

$$\left| n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}_j, \mathbf{Z}_{k,i}) - E(g(\mathbf{w}_j, \mathbf{Z})) \right| \rightarrow 0 \text{ a.s.}$$

As a consequence,

$$\max_{j=1,r} | n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}_j, \mathbf{Z}_{k,i}) - E(g(\mathbf{w}_j, \mathbf{Z})) | \rightarrow 0 \text{ a.s.}$$

Let $\bar{\Lambda} = \bigcap_{\epsilon \in \mathbb{Q}^+} \Lambda_\epsilon$, then $P(\bar{\Lambda}) = 1$. We now prove

$$\begin{aligned} & \forall \omega \in \bar{\Lambda}, \\ & \sup_{\mathbf{w} \in S} | n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}, \mathbf{Z})) | \rightarrow 0. \end{aligned}$$

Let $\eta > 0$, we choose $\epsilon \in \mathbb{Q}^+$ and $\epsilon < \frac{\eta}{3}$. We can recover the inequality (B.2) because the choice mappings are fixed from the beginning. On the other hand, there exists M such that $\forall k \geq M$,

$$\max_{j=1,r} | n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}_j, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}_j, \mathbf{Z})) | < \frac{\eta}{3}.$$

Therefore, for all $k \geq M$,

$$\sup_{\mathbf{w} \in S} | n_k^{-1} \sum_{i=1}^{n_k} g(\mathbf{w}, \mathbf{Z}_{k,i}(\omega)) - E(g(\mathbf{w}, \mathbf{Z})) | < \eta.$$

The proof is completed.

References

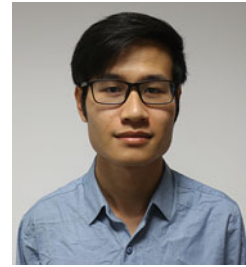
1. Le Thi, H.A., Dinh, T.P.: DC programming and DCA: thirty years of developments. *Mathematical Programming. Special Issue dedicated to: DC Programming - Theory. Algorithms and Appl.* **169**(1), 5–68 (2018)
2. Le Thi, H.A.: DC programming and DCA (2005). <http://www.lita.univ-lorraine.fr/~lethi/index.php/dca.html>
3. Pham Dinh, T., Le Thi, H.A.: The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems. *Ann. Oper. Res.* **133**(1–4), 23–46 (2005)
4. Rockafellar, R.T.: *Convex Analysis*, vol. 28. Princeton University Press, New Jersey (1970)
5. Le Thi, H.A., Le, H.M., Pham Dinh, T.: Feature selection in machine learning: an exact penalty approach using a difference of convex function algorithm. *Mach. Learn.* **101**(1–3), 163–186 (2015)
6. Le Thi, H.A., Le, H.M., Pham Dinh, T., Bouvry, P.: Solving the perceptron problem by deterministic optimization approach based on DC programming and DCA. In: *INDIN*, pp. 222–226. IEEE, Cardiff (2009)
7. Le Thi, H.A., Le, H.M., Pham Dinh, T., Huynh, V.N.: Binary classification via spherical separator by DC programming and DCA. *J. Global Optim.* **56**(4), 1393–1407 (2013)
8. Le Thi, H.A., Moeini, M., Dinh Pham, T.: Portfolio selection under downside risk measures and cardinality constraints based on DC programming and DCA. *Comput. Manag. Sci.* **6**(4), 459–475 (2009)
9. Le Thi, H.A., Moeini, M., Dinh Pham, T.: DC programming approach for portfolio optimization under step increasing transaction costs. *Optimization* **58**(3), 267–289 (2009)
10. Thanh, P.N., Bostel, N., Péton, O.: A DC programming heuristic applied to the logistics network design problem. *Int. J. Prod. Econ.* **135**(1), 94–105 (2012)
11. Ndiaye, B.M., Le Thi, H.A., Pham Dinh, T.: Single straddle carrier routing problem in port container terminals: mathematical model and solving approaches. *Int. J. Intell. Inf. Database Syst.* **6**(6), 532–554 (2012)
12. Le Thi, H.A., Pham Dinh, T.: Difference of convex functions algorithms (DCA) for image restoration via a Markov random field model. *Optim. Eng.* **18**(4), 873–906 (2017)
13. Khalaf, W., Astorino, A., d'Alessandro, P., Gaudioso, M.: A DC optimization-based clustering technique for edge detection. *Optim. Lett.* **11**(3), 627–640 (2017)

14. Le Thi, H.A., Nguyen, Q.T.: A Robust Approach for Nonlinear UAV Task Assignment Problem Under Uncertainty. *Lecture Notes in Computer Science*, vol. 6450, p. 147 (2010)
15. Le, H.M., Le Thi, H.A., Pham Dinh, T., Bouvry, P.: A combined DCA: GA for constructing highly nonlinear balanced boolean functions in cryptography. *J. Global Optim.* **47**(4), 597–613 (2010)
16. Le Thi, H.A., Pham Dinh, T.: A Two Phases DCA Based Algorithm for Solving the Lennard–Jones Problem. LITA, University of Metz, Metz (2011)
17. Le Thi, H.A., Pham Dinh, T., Belghiti, M.: DCA based algorithms for multiple sequence alignment (MSA). *CEJOR* **22**(3), 501–524 (2014)
18. Orlov, A.V., Strekalovsky, A.S.: On a local search for hexamatrix games. In: *DOOR (Supplement)*, pp. 477–488. CEUR-WS, Vladivostok (2016)
19. Le Thi, H.A., Pham Dinh, T.: DC programming in communication systems: challenging problems and methods. *Vietnam J. Comput. Sci.* **1**(1), 15–28 (2014)
20. Liu, Z.: Non-dominated set of a multi-objective optimisation problem. Ph.D. Thesis. Lancaster University, Lancaster (2016)
21. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2**(1), 183–202 (2009)
22. Le Thi, H.A., Moeini, M., Pham Dinh, T.: Portfolio selection under downside risk measures and cardinality constraints based on DC programming and DCA. *Computat. Manag. Sci.* **6**(4), 459–475 (2009)
23. Thiao, M., Pham Dinh, T., Le Thi, H.A.: DC programming approach for a class of nonconvex programs involving l0 norm. In: *International Conference on Modelling, Computation and Optimization in Information Systems and Management Sciences*, pp. 348–357. Springer, Metz/Berlin, Zirndorf (2008)
24. Mordukhovich, B.S.: *Variational Analysis and Generalized Differentiation I: Basic Theory*, vol. 330. Springer Science & Business Media, Dordrecht (2006)
25. Rockafellar, R.T., Wets, R.J.B.: *Variational Analysis*, vol. 317. Springer Science & Business Media, Dordrecht (2009)
26. Pham Dinh, T., Le Thi, H.A.: Convex analysis approach to DC programming: theory, algorithms and applications. *Acta Math. Vietnam.* **22**(1), 289–355 (1997)
27. Pham Dinh, T., Le Thi, H.A.: A DC optimization algorithm for solving the trust-region subproblem. *SIAM J. Optim.* **8**(2), 476–505 (1998)
28. Urruty, J.B.H., Lemaréchal, C.: *Convex Analysis and Minimization Algorithms*. Springer, Berlin (1996)
29. Le Thi, H.A., Huynh, V.N., Pham Dinh, T.: Convergence analysis of difference-of-convex algorithm with subanalytic data. *J. Optimiz. Theory App.* **179**(1), 103–126 (2018)
30. Łojasiewicz, S.: Sur le problème de la division. *Studia Math.* **18**(1), 87–136 (1959)
31. Łojasiewicz, S.: Une propriété topologique des sous-ensembles analytiques réels, Les équations aux dérivées partielles. *Éditions du Centre National de la Recherche Scientifique* **117**, 87–89 (1963)
32. Łojasiewicz, S.: Sur la géométrie semi- et sous- analytique. *Ann. Inst. Fourier* **43**(5), 1575–1595 (1993)
33. Bierstone, E., Milman, P.D.: Semianalytic and subanalytic sets. *IHES Publ. Math.* **67**, 5–42 (1988)
34. Shiota, M.: *Geometry of Subanalytic and Semialgebraic Sets*, vol. 150. Birkhauser, Boston (1997)
35. Bolte, J., Daniilidis, A., Lewis, A.: The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM J. Optim.* **17**(4), 1205–1223 (2007)
36. Ortega, J.M., Rheinboldt, W.C.: *Iterative Solution of Nonlinear Equations in Several Variables*, pp. 253–255. Academic Press, New York (1970)
37. de Leeuw, J.: Applications of convex analysis to multidimensional scaling. In: Barra, J.R., Brodeau, F., Romier, G., Van Cutsem, B. (eds.) *Recent Developments in Statistics*, pp. 133–146. North Holland, Amsterdam (1977)
38. Hunter, D.R., Lange, K.: Rejoinder to discussion of optimization transfer using surrogate objective functions. *J. Comput. Graph. Stat.* **9**, 52–59 (2000)
39. Polyak, B.T.: *Introduction to Optimization*. Optimization Software, Inc., Publications Division, New York (1987)
40. Chambolle, A., De Vore, R.A., Lee, N.Y., Lucier, B.J.: Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Trans. Image Process.* **7**(3), 319–335 (1998)
41. Daubechies, I., Defrise, M., De Mol, C.: An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Commun. Pure Appl. Math.* **57**(11), 1413–1457 (2004)
42. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. B Methodol.* **39**(1), 1–38 (1977)
43. Yuille, A.L., Rangarajan, A.: The concave-convex procedure. *Neural Comput.* **15**(4), 915–936 (2003)
44. Shapiro, A., Dentcheva, D., Ruszczyński, A.: *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, Philadelphia (2009)
45. Schmidt, M., Roux, N.L., Bach, F.R.: Convergence rates of inexact proximal-gradient methods for convex optimization. In: Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F., Weinberger, K.Q. *Proceedings of the 24th NIPS*, pp. 1458–1466. Curran Associates, Inc., Granada (2011)
46. Byrd, R.H., Nocedal, J., Oztoprak, F.: An inexact successive quadratic approximation method for L-1 regularized optimization. *Math. Program.* **157**(2), 375–396 (2016)
47. Fukuda, E.H., Drummond, L.G.: Inexact projected gradient method for vector optimization. *Comput. Optim. Appl.* **54**(3), 473–493 (2013)
48. Martinez, J., Qi, L.: Inexact Newton methods for solving nonsmooth equations. *J. Comput. Appl. Math.* **60**(1-2), 127–145 (1995)
49. Byrd, R.H., Curtis, F.E., Nocedal, J.: An inexact SQP method for equality constrained optimization. *SIAM J. Optimiz.* **19**(1), 351–369 (2008)
50. Gondzio, J.: Convergence analysis of an inexact feasible interior point method for convex quadratic programming. *SIAM J. Optimiz.* **23**(3), 1510–1527 (2013)
51. Leibfritz, F., Sachs, E.M.: Inexact SQP interior point methods and large scale optimal control problems. *SIAM J. Control. Optim.* **38**(1), 272–293 (1999)
52. Robbins, H., Monro, S.: A stochastic approximation method. *Ann. Math. Stat.*, 400–407 (1951)
53. Ermoliev, Y.: Stochastic quasigradient methods and their application to system optimization. *Stochastic* **9**(1-2), 1–36 (1983)
54. Ermol'ev, Y.M., Norkin, V.I.: Stochastic generalized gradient method for nonconvex nonsmooth stochastic optimization. *Cybernet. Systems Anal.* **34**(2), 196–215 (1998)
55. Ermoliev, Y.M., Norkin, V.I.: Solution of nonconvex nonsmooth stochastic optimization problems. *Cybernet. Systems Anal.* **39**(5), 701–715 (2003)
56. Nedic, A., Bertsekas, D.P.: Incremental subgradient methods for nondifferentiable optimization. *SIAM J. Optimiz.* **12**(1), 109–138 (2001)
57. Shalev-Shwartz, S., Singer, Y., Srebro, N., Cotter, A.: Pegasos: primal estimated sub-gradient solver for SVM. *Math. Program.* **127**(1), 3–30 (2011)
58. Nemirovsky, A.S., Yudin, D.B.: *Problem complexity and method efficiency in optimization*. John Wiley & Sons (1983)

59. Bottou, L., Cun, Y.L.: Large scale online learning. In: Thrun, S., Saul, L.K., Schölkopf, B. Proceedings of the 16th NIPS, pp. 217–224. MIT Press, Vancouver and Whistler, British Columbia (2004)
60. Byrd, R.H., Hansen, S.L., Nocedal, J., Singer, Y.: A stochastic quasi-Newton method for large-scale optimization. *SIAM. J. Optimiz.* **26**(2), 1008–1031 (2016)
61. Bordes, A., Bottou, L., Gallinari, P.: SGD-QN: Careful quasi-Newton stochastic gradient descent. *J. Mach. Learn. Res.* **10**(Jul), 1737–1754 (2009)
62. Bottou, L., Bousquet, O.: The tradeoffs of large scale learning. In: Platt, J.C., Koller, D., Singer, Y., Roweis, S.T. (eds.) Proceedings of the 20th NIPS, pp. 161–168. Curran Associates, Inc., Vancouver (2008)
63. Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.* **12**(Jul), 2121–2159 (2011)
64. Wang, X., Ma, S., Goldfarb, D., Liu, W.: Stochastic quasi-Newton methods for nonconvex stochastic optimization. *SIAM. J. Optimiz.* **27**(2), 927–956 (2017)
65. Roux, N.L., Fitzgibbon, A.W.: A fast natural Newton method. In: Furnkranz, J., Joachims, T. (eds.) Proceedings of the 27th International Conference on Machine Learning, pp. 623–630. Omnipress, Haifa (2010)
66. Mokhtari, A., Ribeiro, A.: RES: Regularized stochastic BFGS algorithm. *IEEE Trans. Signal Process.* **62**(23), 6089–6104 (2014)
67. Mokhtari, A., Ribeiro, A.: Global convergence of online limited memory BFGS. *J. Mach. Learn. Res.* **16**(1), 3151–3181 (2015)
68. Roux, N.L., Schmidt, M., Bach, F.R.: A stochastic gradient method with an exponential convergence rate for finite training sets. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Proceedings of the 25th NIPS, pp. 2663–2671. Curran Associates, Inc., Nevada (2012)
69. Blatt, D., Hero, A.O., Gauchman, H.: A convergent incremental gradient method with a constant step size. *SIAM. J. Optimiz.* **18**(1), 29–51 (2007)
70. Ghadimi, S., Lan, G.: Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization I: a generic algorithmic framework. *SIAM. J. Optimiz.* **22**(4), 1469–1492 (2012)
71. Nesterov, Y.: Lectures on Convex Optimization, vol. 137. Springer, Cham (2018)
72. Shalev-Shwartz, S., Zhang, T.: Stochastic dual coordinate ascent methods for regularized loss minimization. *J. Mach. Learn. Res.* **14**(Feb), 567–599 (2013)
73. Shalev-Shwartz, S., Zhang, T.: Accelerated mini-batch stochastic dual coordinate ascent. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. Proceedings of the 26th NIPS, pp. 378–385. Curran Associates, Inc., Nevada (2013)
74. Shalev-Shwartz, S., Zhang, T.: Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. In: Xing, E.P., Jebara, T. (eds.) Proceedings of the 31st International Conference on Machine Learning, pp. 64–72. PMLR, Beijing (2014)
75. Johnson, R., Zhang, T.: Accelerating stochastic gradient descent using predictive variance reduction. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. Advances in Neural Information Processing Systems, pp. 315–323. Curran Associates, Inc., Nevada (2013)
76. Xiao, L., Zhang, T.: A proximal stochastic gradient method with progressive variance reduction. *SIAM. J. Optimiz.* **24**(4), 2057–2075 (2014)
77. Nitanda, A.: Stochastic proximal gradient descent with acceleration techniques. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Proceedings of the 27th NIPS, pp. 1574–1582. Curran Associates, Inc., Montreal (2014)
78. Juditsky, A., Nemirovski, A.: First order methods for nonsmooth convex large-scale optimization, I: general purpose methods. *Optim. Mach. Learn.*, 121–148 (2011)
79. Duchi, J.C., Shalev-Shwartz, S., Singer, Y., Tewari, A.: Composite objective mirror descent. In: Kalai, A.T., Mohri, M. Conference on Learning Theory, pp. 14–26. Omnipress, Haifa (2010)
80. Nemirovski, A., Juditsky, A., Lan, G., Shapiro, A.: Robust stochastic approximation approach to stochastic programming. *SIAM. J. Optimiz.* **19**(4), 1574–1609 (2009)
81. Mairal, J.: Incremental majorization-minimization optimization with application to large-scale machine learning. *SIAM. J. Optimiz.* **25**(2), 829–855 (2015)
82. Duchi, J.C., Ruan, F.: Stochastic methods for composite and weakly convex optimization problems. *SIAM. J. Optimiz.* **28**(4), 3229–3259 (2018)
83. Davis, D., Grimmer, B.: Proximally guided stochastic subgradient method for nonsmooth, nonconvex problems. *SIAM. J. Optimiz.* **29**(3), 1908–1930 (2019)
84. Le Thi, H.A., Le, H.M., Phan, D.N., Tran, B.: Stochastic DCA for the large-sum of non-convex functions problem and its application to group variable selection in classification. In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 3394–3403. PMLR, Sydney (2017)
85. Le Thi, H.A., Le, H.M., Phan, D.N., Tran, B.: Stochastic DCA for minimizing a large sum of DC functions with application to Multi-class Logistic Regression (2019). arXiv:1911.03992
86. Le Thi, H.A., Huynh, V.N., Pham Dinh, T.: Stochastic Difference-of-Convex Algorithms for solving nonconvex optimization problems (2019). arXiv:1911.04334
87. Nitanda, A., Suzuki, T.: Stochastic difference of convex algorithm and its application to training deep Boltzmann machines. In: Singh, A., Zhu, J. (eds.) Proceedings of the 20th International Conference Artificial Intelligence and Statistics, pp. 470–478. PMLR, Fort Lauderdale (2017)
88. Xu, Y., Qi, Q., Lin, Q., Jin, R., Yang, T.: Stochastic optimization for DC functions and non-smooth non-convex regularizers with non-asymptotic convergence. In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of the 36th International Conference on Machine Learning, pp. 6942–6951. PMLR, Long Beach (2019)
89. Liu, J., Cui, Y., Pang, J.S., Sen, S.: Two-stage stochastic programming with linearly Bi-parameterized quadratic recourse. Preprint (2019)
90. Allen-Zhu, Z., Hazan, E.: Variance reduction for faster non-convex optimization. In: Balcan, M.F., Weinberger, K.Q. (eds.) Proceedings of the 33rd International Conference Machine Learning, pp. 699–707. PMLR, New York (2016)
91. Dupacová, J., Wets, R.: Asymptotic behavior of statistical estimators and of optimal solutions of stochastic optimization problems. *Ann. Stat.* **16**(4), 1517–1549 (1988)
92. Durrett, R.: Probability: Theory and Examples, vol. 49, Chap. 5. Cambridge University Press, New York (2019)
93. Métivier, M.: Semimartingales: A Course on Stochastic Processes, vol. 2, , Chap. 2, pp. 49–51. Bauer, H., Gabriel, P. (eds.) Walter de Gruyter, Berlin (1982)
94. Yuan, K.H.: A theorem on uniform convergence of stochastic functions with applications. *J. Multivar. Anal.* **62**(1), 100–109 (1997)
95. Fonseca, I., Leoni, G.: Modern Methods in the Calculus of Variations: L^p Spaces, pp. 150–151. Springer Science & Business Media, New York (2007)
96. Wen, B., Chen, X., Pong, T.K.: A proximal difference-of-convex algorithm with extrapolation. *Comput. Optim. Appl.* **69**(2), 297–324 (2018)



Hoai An Le Thi An obtained her PhD with Distinction in Optimization in 1994, her Habilitation in 1997 from university of Rouen, France. She is currently Professor exceptional class, university of Lorraine. She is the author/co-author of more than 240 papers, the co-editor of 22 books/journal special issues, supervisor of 30 PhD theses. Her research interests include Machine Learning, Optimization, Operations Research and their applications.



Luu Hoang Phuc Hau is currently a Ph.D. student working at the University of Lorraine. His research interests include deterministic optimization and stochastic optimization.



Tao Pham Dinh earned his Doctor of Sciences on Numerical Analysis and Optimization in 1981 from University Joseph Fourier-Grenoble I, France. He is Full Professor at National Institute for Applied Sciences (INSA)-Rouen, France. He is the author of more than 220 papers, editor of 7 books and 10 journal Special Issues, supervisor of 50 PhD theses. His research interests include Numerical Analysis, Optimization and Operations Research and their applications.



Hoai Minh Le received his PhD in Computer Science from University Paul Verlaine-Metz, France in 2007. He worked at University of Le Havre, at Luxembourg Centre for Systems Biomedicine, and at Computer Science and Applications Department, LGIPM, University of Lorraine, France. His research interests include non-convex optimization and its applications in industrial systems, Data Mining and Machine Learning.



Inference for Coherent Systems with Weibull Components Under a Simple Step-Stress Model

35

Narayanaswamy Balakrishnan, Debanjan Mitra, and Xiaojun Zhu

Contents

35.1	Introduction	703
35.2	A Simple Step-Stress Model for Coherent Systems	705
35.3	Likelihood Inference	706
35.3.1	Direct Maximization of the Observed Likelihood.....	706
35.3.2	Estimation Using the St-EM Algorithm.....	709
35.3.3	Providing Initial Values to the Algorithm.....	710
35.3.4	Confidence Intervals.....	711
35.4	Outline of Bayesian Analysis	712
35.5	Simulation Study	713
35.6	Illustrative Example	729
35.7	Concluding Remarks	729
	References	729

Abstract

Coherent systems are widely studied in reliability experiments. Under the assumption that the components of a coherent system follow a two-parameter Weibull distribution, maximum likelihood inference for n -component coherent systems with known signatures under a simple step-stress model is discussed in this paper. The detailed steps of the stochastic expectation maximization algorithm under this setup are also developed to obtain estimates of the model parameters. Asymptotic confidence intervals for

the model parameters are constructed using the observed Fisher information matrix and missing information principle. Parametric bootstrap approach is used also to construct confidence intervals for the parameters. A method based on best linear unbiased estimators is developed to provide initial values that are needed for numerical computation of maximum likelihood estimates. The performance of the methods developed is assessed through an extensive Monte Carlo simulation study. Finally, two numerical examples are presented for illustrative purpose.

Keywords

Coherent systems · Best linear unbiased estimators · Bootstrap method · Confidence intervals · Maximum likelihood estimates · Stochastic expectation maximization algorithm · Missing information principle · Step-stress model · Total time on test transforms · Type-II censoring · Weibull distribution

35.1 Introduction

Due to their high reliability, the products that are tested in different industrial experiments often have very large mean failure times under normal operating conditions. As a result, very few failures, if any, are observed in life testing experiments involving such products. Hence, due to lack of sufficient amount of information, efficient inference regarding the lifetimes of such units cannot be made even after employing sophisticated statistical methods for Type-I or Type-II censored data. One way out for this problem is to use accelerated testing procedures, wherein the units are subjected to higher stress level than normal, in order to accelerate their failures. For example, in order to assess the effect of temperature, load, voltage, etc., on the lifetimes of engineering products, an experimenter may use accelerated tests, which may then be performed using constant stress

N. Balakrishnan (✉)
Department of Mathematics and Statistics, McMaster University,
Hamilton, ON, Canada
e-mail: bala@mcmaster.ca

D. Mitra
Indian Institute of Management Udaipur, Udaipur, Rajasthan, India
e-mail: debanjan.mitra@iimu.ac.in

X. Zhu
Xi'an Jiaotong-Liverpool University, Suzhou, China
e-mail: Xiaojun.Zhu@xjtlu.edu.cn

or linearly increasing stress levels. Once the experiment accelerates the failures of the experimental units, the obtained data may then be extrapolated using appropriate techniques to estimate the underlying failure time distribution under normal operating conditions. This process requires a model relating the stress level and the failure time distribution. For details, one may see Nelson [1, 2], Nelson and Meeker [3], Meeker and Hahn [4], Meeker and Escobar [5], and Bagdonavicius and Nikulin [6].

The step-stress experiment is a special case of accelerated testing. Here, under an initial stress level, say s_0 , n identical units are placed on a life test. Then, at pre-fixed times $\tau_1, \tau_2, \dots, \tau_k$, the stress levels are increased to s_1, s_2, \dots, s_k , respectively. Under this condition, one may then fix Type-I or Type-II censoring scheme for termination of the experiment, and suitable methods may then be applied to make necessary inference. A simple step-stress model is the one with only two stress levels; it has been studied extensively in the literature. For example, Miller and Nelson [7] and Bai et al. [8] determined the optimal time at which the stress level is changed from the first to second; DeGroot and Goel [9] developed a Bayesian framework for optimal tests. More recently, the simple-stress experiment has been discussed extensively under different censoring schemes, for different lifetime models; see Balakrishnan et al. [10], Balakrishnan [11], Balakrishnan and Xie [12, 13], Han and Balakrishnan [14], Kateri and Balakrishnan [15], and the references therein.

Coherent systems in reliability analysis can be quite complex sometimes, having many different components with many different connections. The failure of such a system, clearly, depends on the failure of one or more of its components. The system signature of a coherent system is a purely distribution-free measure that has been widely used in the literature to develop statistical inference regarding system as well as component lifetime characteristics. For details regarding system signature-based statistical inference for coherent systems, see Samaniego [16] and Navarro et al. [17], among others. Recently, Balakrishnan et al. [18, 19] have analyzed the lifetime distribution of the components of coherent systems with known system signatures based on system-level failure time data, through linear and nonparametric statistical inferential techniques.

Suppose 200 components are placed in a simple step-stress life test with the change of stress at time τ . Instead of such a step-stress test on components, one may construct a step-stress life test based on some systems that have been formed with these components. For example, Table 35.1 shows the first 30 failures from 50 4-component coherent systems with the change of stress done at time $\tau=10$ hours. In this example, the probability a system will breakdown is $0, 1/2, 1/4$, and $1/4$ after the failure of the 1st, 2nd, 3rd, and 4th components, respectively. Then, it will be of interest to develop the estimation of model parameters based on such a system lifetime data. These estimates can be then used to estimate some reliability characteristics of systems and components as well. We may also be interested in comparing the efficiency of the tests based on systems instead of on components directly. Here, we assume the components are Weibull distributed and the signature of the system under consideration is known.

In this paper, we develop statistical inference for the lifetime distribution of components of coherent systems with known signatures based on system-level failure time data, when the coherent systems are put on a simple step-stress experiment. The components of a coherent system are assumed to follow, independently and identically, a Weibull distribution, which is a widely used lifetime model. In this regard, it may be mentioned that Kateri and Balakrishnan [15] developed inference for a simple step-stress model with Weibull distributed lifetimes, under Type-II censoring. Here, we develop likelihood inference for the model parameters under the abovementioned setup. We also develop asymptotic confidence intervals (CIs) for the parameters.

It can be easily understood that for such a complex model, explicit expressions for the maximum likelihood estimates (MLEs) cannot be obtained, and one has to use a numerical technique. We take two different approaches to estimate the parameters. The first approach is the direct method of maximizing the observed likelihood by Newton-Raphson method to obtain the numerical MLEs. The other approach is based on the stochastic expectation-maximization (St-EM) algorithm, for which we develop the problem from the perspective of incomplete data. The St-EM algorithm, originally proposed by Celeux and Diebolt [20], is a variant of the expectation-maximization (EM) algorithm proposed

Table 35.1 Failure time data of four-component coherent systems with signature vector $\mathbf{P} = (0, 1/2, 1/4, 1/4)$

Stress level	Failure times					
First	6.3767	7.1181	7.5412	8.1062	8.6519	8.8920
	9.4739	9.7148				
Second	10.0610	10.1218	10.2550	10.2824	10.4467	10.5271
	10.7169	10.7643	10.7677	10.8292	10.8777	10.9346
	11.0665	11.0897	11.1372	11.1797	11.1986	11.2969
	11.3438	11.4184	11.4482	11.4770		

by Dempster et al. [21]. While the EM algorithm is a very powerful tool for analyzing incomplete data, the algorithm may get very complex sometimes when the required conditional expectations are difficult to obtain analytically. Moreover, it is well-known that the EM algorithm suffers from the existence of local optima [22]. The St-EM algorithm is one of the many variants of the EM algorithm that have been proposed to overcome the difficulties of the original EM algorithm. We develop the steps of the St-EM algorithm under this setup, to estimate the parameters.

For a numerical estimation of the parameters, one needs to give carefully chosen initial values, especially given the complex nature of the likelihood function to be optimized. We propose a technique to provide reasonable initial values for the algorithm. Then, through an extensive Monte Carlo simulation study, we examine the performance of the proposed methods of statistical inference.

The rest of this paper is organized as follows. In Sect. 35.2, we discuss the structure of the data and present the model. The likelihood function, derivation of the MLEs, and asymptotic CIs for the parameters are discussed in Sect. 35.3. Formulating this problem as an incomplete data problem, we also present the steps of the St-EM algorithm under this setup in this section and discuss construction of asymptotic confidence intervals for the model parameters within the framework of the St-EM algorithm. In addition, this section also contains a discussion on the choice of initial values. Finally, construction of asymptotic confidence intervals for the model parameters by the observed Fisher information matrix, missing information principle, and parametric bootstrap technique are discussed in this section. We present an outline of Bayesian analysis under this setup in Sect. 35.4. In Sect. 35.5, we present the numerical results of a detailed simulation study to assess and compare the performances of the inferential methods presented in this paper. A numerical example for illustrative purpose is presented in Sect. 35.6. Finally, we conclude the paper with some comments in Sect. 35.7.

35.2 A Simple Step-Stress Model for Coherent Systems

Let there be m coherent systems, each with n components. Let T denote the lifetime of a coherent system, and let X_1, \dots, X_n denote the lifetimes of its components, which are assumed to be independently and identically (i.i.d.) distributed with probability density function (PDF) $f_X(\cdot)$ and cumulative distribution function (CDF) $F_X(\cdot)$. Let $S_X(\cdot)$ denote the survival function (SF) of the component lifetime distribution.

Then, the system signature of a coherent system is given by \mathbf{p} , where $\mathbf{p} = (p_1, \dots, p_n)'$ with

$$p_i = P(T = X_{i:n}), \quad i = 1, 2, \dots, n,$$

$X_{i:n}$ being the i -th smallest order statistic arising out of X_1, \dots, X_n . Clearly, we have

- (i) $p_i \geq 0, \forall i$,
- (ii) p_i 's do not depend on the component lifetime distribution, and
- (iii) $\sum_{i=1}^n p_i = 1$;

see Samaniego [16] for a detailed discussion on system signatures.

Equivalent to the system signature, one can define the minimal signature $\mathbf{a} = (a_1, \dots, a_n)$, with a_i 's being negative and non-negative quantities satisfying $\sum_{i=1}^n a_i = 1$; see Navarro et al. [23]. The system signature \mathbf{p} and minimal signature \mathbf{a} are related by

$$a_i = \frac{1}{i} \sum_{j=1}^n j(-1)^{i+j-n-1} \binom{j-1}{n-i} p_j \binom{n}{j}, \quad i = 1, 2, \dots, n.$$

The PDF $f_T(\cdot)$ and SF $S_T(\cdot)$ of the system lifetime T can be expressed as a mixture of distributions of order statistics arising from the component lifetimes in the system as follows (see Samaniego [24] and Kochar et al. [25]):

$$f_T(t) = \sum_{i=1}^n p_i f_{i:n}(t),$$

$$S_T(t) = \sum_{i=1}^n p_i S_{i:n}(t),$$

where $f_{i:n}(\cdot)$ and $S_{i:n}(\cdot)$ are the PDF and SF of the i -th order statistic $X_{i:n}$ arising out of X_1, \dots, X_n , given by

$$f_{i:n}(t) = \binom{n}{i} i f_X(t) [F_X(t)]^{i-1} [S_X(t)]^{n-i},$$

$$S_{i:n}(t) = \sum_{j=0}^{i-1} \binom{n}{j} [F_X(t)]^j [S_X(t)]^{n-j};$$

see Arnold et al. [26].

Using minimal signature, similarly, the distribution of T can be expressed as a generalized mixture of distributions of order statistics arising from the component lifetimes. Precisely, the PDF and SF of T can be expressed using the minimal signature as (see Navarro et al. [23])

$$f_T(t) = \sum_{i=1}^n a_i f_{1:i}(t) = \sum_{i=1}^n a_i i f_X(t) [S_X(t)]^{i-1},$$

$$S_T(t) = \sum_{i=1}^n a_i S_{1:i}(t) = \sum_{i=1}^n a_i [S_X(t)]^i.$$

Suppose $m n$ -component coherent systems are placed on a simple step-stress experiment with stress levels s_1 and s_2 and with τ being the time point at which the stress level changes from s_1 to s_2 . In practice, the experimenter will often have failure time data at the system level, i.e., he/she observes failure times of systems, rather than their components. Also, due to time constraint on the test, one always observes censored data. We assume Type-II censoring, or failure censoring, here; that is, the experiment is terminated when r system failures are observed. Let n_1 be the random number of system failures observed before time τ (i.e., at stress level s_1) and $n_2 = r - n_1$ be the random number of system failures observed after time τ (i.e., at stress level s_2). The observed Type-II censored data are then represented by the ordered system failure times

$$t_{1:m} < t_{2:m} < \dots < t_{r:m}.$$

As mentioned earlier, in a step-stress experiment, a model relates the stress level and the failure time distribution. Here, we assume the well-known cumulative exposure model; see Nelson [2]. We assume that the lifetime of a component of a system follows Weibull distribution with scale parameters θ_1 and θ_2 and a common shape parameter β , under stress levels s_1 and s_2 , respectively. Then, under the assumption of cumulative exposure model, the CDF of the lifetime of a component of a system is given by

$$F_X(t) = \begin{cases} F_1(t), & 0 < t < \tau, \\ F_2\left(\frac{\theta_2}{\theta_1}\tau + t - \tau\right), & \tau \leq t < \infty, \end{cases} \quad (35.1)$$

where $F_j(x) = 1 - e^{-(x/\theta_j)^\beta}$, with $x > 0$, $\theta_j > 0$, $\beta > 0$ $j = 1, 2$. Correspondingly, the PDF of a component is given by

$$f_X(t) = \begin{cases} \frac{\beta}{\theta_1^\beta} t^{\beta-1} e^{-(t/\theta_1)^\beta}, & 0 < t < \tau, \\ \frac{\beta}{\theta_2^\beta} \left(\frac{\theta_2}{\theta_1}\tau + t - \tau\right)^{\beta-1} e^{-\left(\frac{\theta_2}{\theta_1}\tau + t - \tau\right)^\beta}, & \tau \leq t < \infty. \end{cases} \quad (35.2)$$

The following propositions are necessary for the completeness of the model and the subsequent developments on inference.

Proposition 2.1 *Suppose the component lifetimes are Weibull distributed. Then, if a simple step-stress life test is conducted on a coherent system with lifetime T under the cumulative exposure model, it is equivalent to conducting a simple step-stress test on its components with lifetimes X_i 's, under the cumulative exposure model.*

Proof. If $F_{T,1}(\cdot)$ and $F_{T,2}(\cdot)$ are the CDFs of the system lifetime T under the first and second stress levels, respectively, then from (35.1) we have

$$\begin{aligned} F_T(t) &= \begin{cases} F_{T,1}(t), & 0 < t < \tau, \\ F_{T,2}\left(\frac{\theta_2}{\theta_1}\tau + t - \tau\right), & \tau \leq t < \infty, \end{cases} \\ &= \begin{cases} \sum_{i=1}^n p_i F_{i:n}(t), & 0 < t < \tau, \\ \sum_{i=1}^n p_i F_{i:n}\left(\frac{\theta_2}{\theta_1}\tau + t - \tau\right), & \tau \leq t < \infty, \end{cases} \\ &= \begin{cases} \sum_{i=1}^n p_i \sum_{j=i}^n \binom{n}{j} [F_X(t)]^j [S_X(t)]^{n-j}, & 0 < t < \tau, \\ \sum_{i=1}^n p_i \sum_{j=i}^n \binom{n}{j} \left[F_X\left(\frac{\theta_2}{\theta_1}\tau + t - \tau\right)\right]^j \\ \left[S_X\left(\frac{\theta_2}{\theta_1}\tau + t - \tau\right)\right]^{n-j}, & \tau \leq t < \infty. \end{cases} \end{aligned}$$

From above, the required result follows. ■

Proposition 2.2 *Under a simple step-stress life testing experiment on a coherent system with lifetime T and a Weibull component lifetime distribution as in (35.2), the SF of T is a non-decreasing function of θ_2 .*

Proof. Evidently,

$$\frac{\partial S_T(t; \theta_1, \theta_2, \beta)}{\partial \theta_2} = \begin{cases} 0, & 0 < t < \tau, \\ \frac{t-\tau}{\theta_2} f_T(t, \theta_1, \theta_2, \beta) > 0, & \tau \leq t < \infty, \end{cases}$$

which proves the required result. ■

35.3 Likelihood Inference

35.3.1 Direct Maximization of the Observed Likelihood

The statistical inference developed here assumes that the system signatures (or equivalently, minimal signatures) are available. Based on the observed ordered Type-II censored lifetimes of systems $t_{1:m} < \dots < t_{r:m}$ arising out of the m systems, the likelihood function can be written for different values of the random number of failures n_1 : (i) $n_1 = r$, i.e., all the r failures are observed in the first stress level s_1 , (ii) $n_1 = 0$, i.e., all the r failures are observed in the second stress level s_2 , and (iii) $1 \leq n_1 \leq r - 1$. We present the likelihood function below for these three cases under a simple step-stress model:

Case (i): $n_1 = r$

$$L(\theta) = C \prod_{k=1}^r \left[\sum_{i=1}^n a_i f_X(t_{k:m}) \{S_X(t_{k:m})\}^{i-1} \right] \left[\sum_{i=1}^n a_i \{S_X(t_{r:m})\}^i \right]^{m-r};$$

Case (ii): $n_1 = 0$

$$L(\theta) = C \prod_{k=1}^r \left[\sum_{i=1}^n a_i f_X(t_{k:m}) \{S_X(t_{k:m})\}^{i-1} \right] \left[\sum_{i=1}^n a_i \{S_X(t_{r:m})\}^i \right]^{m-r};$$

Case (iii): $1 \leq n_1 \leq r - 1$

$$L(\theta) = C \prod_{k=1}^{n_1} \left[\sum_{i=1}^n a_i f_X(t_{k:m}) \{S_X(t_{k:m})\}^{i-1} \right] \times \prod_{k=n_1+1}^r \left[\sum_{i=1}^n a_i f_X(t_{k:m}) \{S_X(t_{k:m})\}^{i-1} \right] \left[\sum_{i=1}^n a_i \{S_X(t_{r:m})\}^i \right]^{m-r},$$

where $C = \frac{m!}{(m-r)!}$ and θ is the parameter vector of the component lifetime distribution. Since we have assumed that the components of the systems are Weibull distributed, the likelihood function takes the following forms for the three cases; here, we have $\theta = (\theta_1, \theta_2, \beta)$:

Case (i): $n_1 = r$

$$L(\theta) = C \left(\frac{\beta}{\theta_1^\beta} \right)^r \prod_{k=1}^r \left[t_{k:m}^{\beta-1} \sum_{i=1}^n a_i i e^{-i(t_{k:m}/\theta_1)^\beta} \right] \left[\sum_{i=1}^n a_i e^{-i(t_{r:m}/\theta_1)^\beta} \right]^{m-r}; \quad (35.3)$$

Case (ii): $n_1 = 0$

$$L(\theta) = C \left(\frac{\beta}{\theta_2^\beta} \right)^r \prod_{k=1}^r \left[\left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^{\beta-1} \sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^\beta} \right] \times \left[\sum_{i=1}^n a_i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right)^\beta} \right]^{m-r}; \quad (35.4)$$

Case (iii): $1 \leq n_1 \leq r - 1$

$$L(\theta) = C \left(\frac{\beta}{\theta_1^\beta} \right)^{n_1} \prod_{k=1}^{n_1} \left[t_{k:m}^{\beta-1} \sum_{i=1}^n a_i i e^{-i(t_{k:m}/\theta_1)^\beta} \right]$$

$$\times \left(\frac{\beta}{\theta_2^\beta} \right)^{r-n_1} \prod_{k=n_1+1}^r \left[\left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^{\beta-1} \sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^\beta} \right] \times \left[\sum_{i=1}^n a_i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right)^\beta} \right]^{m-r}. \quad (35.5)$$

Note that MLE of θ_2 does not exist in Case (i). So, we shall consider the case $0 \leq n_1 \leq r - 1$ to obtain the MLEs; note that this includes both Cases (ii) and (iii). The log-likelihood function becomes

$$\begin{aligned} \log L(\theta) &= C + r \log \beta - n_1 \beta \log \theta_1 - (r - n_1) \beta \log \theta_2 \\ &+ \sum_{k=1}^{n_1} \left[(\beta - 1) \log t_{k:m} \right. \\ &+ \left. \log \left(\sum_{i=1}^n a_i i e^{-i(t_{k:m}/\theta_1)^\beta} \right) \right] \\ &+ \sum_{k=n_1+1}^r \left[(\beta - 1) \log \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right) \right. \\ &+ \left. \log \left(\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^\beta} \right) \right] \\ &+ (m - r) \log \left(\sum_{i=1}^n a_i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right)^\beta} \right). \quad (35.6) \end{aligned}$$

From the log-likelihood function in (35.6), we get the first-order derivatives as follows:

$$\begin{aligned} \frac{\partial \log L}{\partial \theta_1} &= -\frac{n_1 \beta}{\theta_1} + \sum_{k=1}^{n_1} \left[\frac{\beta t_{k:m}^\beta \sum_{i=1}^n a_i i^2 e^{-i(t_{k:m}/\theta_1)^\beta}}{\theta_1^{\beta+1} \sum_{i=1}^n a_i i e^{-i(t_{k:m}/\theta_1)^\beta}} \right] \\ &- \frac{\theta_2 \tau}{\theta_1^2} \sum_{k=n_1+1}^r \left[\frac{\beta - 1}{\left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)} \right. \\ &- \left. \frac{\beta}{\theta_2^\beta} \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^{\beta-1} \frac{\sum_{i=1}^n a_i i^2 e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^\beta}}{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^\beta}} \right] \end{aligned}$$

$$+ (m-r) \frac{\beta}{\theta_2^\beta} \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right)^{\beta-1} \frac{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right) / \theta_2}^\beta}{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right) / \theta_2}^\beta} \right], \quad (35.7)$$

$$\begin{aligned} \frac{\partial \log L}{\partial \theta_2} &= -\frac{(r-n_1)\beta}{\theta_2} + \sum_{k=n_1+1}^r \left[\frac{(\beta-1)\frac{\tau}{\theta_1}}{\left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)} + \beta \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^{\beta-1} \right. \\ &\quad \times \left. \left(\frac{t_{k:m} - \tau}{\theta_2^2} \right) \frac{\sum_{i=1}^n a_i i^2 e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right) / \theta_2}^\beta}{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right) / \theta_2}^\beta} \right] \\ &\quad + (m-r)\beta \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right)^{\beta-1} \left(\frac{t_{r:m} - \tau}{\theta_2^2} \right) \\ &\quad \times \frac{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right) / \theta_2}^\beta}{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right) / \theta_2}^\beta}, \end{aligned} \quad (35.8)$$

$$\begin{aligned} \frac{\partial \log L}{\partial \beta} &= \frac{r}{\beta} - n_1 \log \theta_1 - (r-n_1) \log \theta_2 \\ &\quad + \sum_{k=1}^{n_1} \left[\log t_{k:m} - \left(\frac{t_{k:m}}{\theta_1} \right)^\beta \log \left(\frac{t_{k:m}}{\theta_1} \right) \frac{\sum_{i=1}^n a_i i^2 e^{-i(t_{k:m}/\theta_1)^\beta}}{\sum_{i=1}^n a_i i e^{-i(t_{k:m}/\theta_1)^\beta}} \right] \\ &\quad + \sum_{k=n_1+1}^r \left[\log \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right) - \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right)^\beta \right. \\ &\quad \times \left. \log \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right) \frac{\sum_{i=1}^n a_i i^2 e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right) / \theta_2}^\beta}{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{k:m} - \tau \right) / \theta_2}^\beta} \right] \\ &\quad - (m-r) \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right)^\beta \log \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right) \\ &\quad \times \frac{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right) / \theta_2}^\beta}{\sum_{i=1}^n a_i i e^{-i \left(\frac{\theta_2}{\theta_1} \tau + t_{r:m} - \tau \right) / \theta_2}^\beta}. \end{aligned} \quad (35.9)$$

Equating expressions (35.7)–(35.9) to zero and then solving these likelihood equations, we get the MLEs. It is clear from the expressions in (35.7)–(35.9) that the MLEs are not available in explicit form, and one has to employ numerical techniques, such as Newton-Raphson method, to obtain the MLE, which we denote by $\hat{\theta}$. For an iterative algorithm like Newton-Raphson, at each stage i we get an estimate $\hat{\theta}_{(i)}$ and stop at a stage l when two successive estimates at stages l and $l - 1$ are such that $|\hat{\theta}_{(l)} - \hat{\theta}_{(l-1)}| < \epsilon$, for a pre-fixed $\epsilon > 0$.

35.3.2 Estimation Using the St-EM Algorithm

Because of its stochastic nature, the St-EM algorithm can avoid some of the common pitfalls of the traditional EM algorithm. For example, to implement St-EM algorithm, it is not required to obtain the complex conditional expectations that may arise in an incomplete data problem; only the appropriate conditional distributions need to be identified. Also, the sequence of estimates obtained through St-EM algorithm does not get trapped into saddle points, if they exist, unlike the EM algorithm [27]. After it was proposed by Celeux and Diebolt [20], many researchers have investigated the St-EM algorithm in further detail and have applied it successfully to various incomplete data problems; see Celeux et al [28], Chauveau [29], Diebolt and Celeux [30], and Nielsen [31].

Broadly speaking, the St-EM algorithm has two steps – the stochastic expectation (St-E) step and the maximization (M) step. Given the observed data and current values of the parameters, random observations are generated from appropriate conditional distributions to replace the unobserved (or censored) data in the St-E step. Then, with the pseudo-complete data thus obtained, the pseudo-complete likelihood is constructed and is maximized to obtain updated estimates of the parameters in the M-step. The St-E and the M-steps are then iterated for sufficiently large number of times.

To implement St-EM algorithm, one needs to construct the pseudo-complete likelihood with the pseudo-complete data as obtained from the St-E step. To construct the pseudo-complete likelihood, in this section we follow the earlier formulation of the problem as detailed in Sect. 35.3.1. Keeping the symbols unchanged, we need to consider the following three cases: (i) $n_1 = r$, (ii) $n_1 = 0$, and (iii) $1 \leq n_1 \leq r - 1$. For these three cases, the pseudo-complete likelihood is as follows:

Case (i): $n_1 = r$

$$L_{PC}(\theta) = \prod_{j=1}^m \left\{ \sum_{i=1}^n a_i f_X(t_{j:m}) \{S_X(t_{j:m})\}^{i-1} \right\}$$

Case (ii): $n_1 = 0$

$$L_{PC}(\theta) = \prod_{j=1}^m \left\{ \sum_{i=1}^n a_i f_X(t_{j:m}) \{S_X(t_{j:m})\}^{i-1} \right\}$$

Case (iii): $1 \leq n_1 \leq r - 1$

$$L_{PC}(\theta) = \prod_{j=1}^{n_1} \left\{ \sum_{i=1}^n a_i f_X(t_{j:m}) \{S_X(t_{j:m})\}^{i-1} \right\} \times \prod_{j=n_1+1}^m \left\{ \sum_{i=1}^n a_i f_X(t_{j:m}) \{S_X(t_{j:m})\}^{i-1} \right\}$$

The St-EM algorithm under this setup takes the following shape:

Algorithm: At the l -th stage of the algorithm:

The St-E step:

- (1) Based on the value of n_1 , generate $m - r$ system lifetimes from the appropriate conditional distribution with CDF $F_T(t|r_{r:m}, \theta^l)$, where θ is the appropriate parameter vector.
- (2) Using the generated data, order the pseudo-complete data $t_{1:m} < \dots < t_{r:m} < t_{r+1:m}^g < \dots < t_{m:m}^g$, where the superscript (g) indicates the randomly generated observations.
- (3) Using the pseudo-complete data, construct the pseudo-complete likelihood depending on the value of n_1 , according to the cases (i), (ii), and (iii) as mentioned above.

The M-step: Maximize the pseudo-complete likelihood as obtained the St-E step to obtain updated estimate of the parameter vector, i.e., θ^{l+1} .

The St-E and M-steps are then repeated for N number of times, to obtain sequences of estimates for the parameters, i.e., $(\hat{\theta}_{11}, \hat{\theta}_{12}, \dots, \hat{\theta}_{1N})$, $(\hat{\theta}_{21}, \hat{\theta}_{22}, \dots, \hat{\theta}_{2N})$, and $(\hat{\beta}_1, \hat{\beta}_2, \dots, \hat{\beta}_N)$. Finally, after discarding first N_1 estimates for each of them for burn-in, we select every 5th value from these sequences starting from $(N_1 + 1)$ -th value till the N -th value and average over the selected values to get the final estimates of θ_1 , θ_2 and β . Every 5th value starting from the $(N_1 + 1)$ th value is selected to eliminate the auto-correlation, if any, among the successive values.

The conditional distributions required in the St-E step are as follows:

Case (i): $n_1 = r$

$$F_T(t|r_{r:m}, \theta) = \frac{F_{T,1}(t) - F_{T,1}(t_{r:m})}{1 - F_{T,1}(t_{r:m})}, \quad t > t_{r:m}, \quad (35.10)$$

Case (ii): $n_1 = 0$

$$F_T(t|r_{r:m}, \theta) = \frac{F_{T,2}(t) - F_{T,2}(t_{r:m})}{1 - F_{T,2}(t_{r:m})}, \quad t > t_{r:m}, \quad (35.11)$$

Case (iii): $1 \leq n_1 \leq r - 1$

$$F_T(t|t_{r:m}, \theta) = \frac{F_{T,2}(t) - F_{T,2}(t_{r:m})}{1 - F_{T,2}(t_{r:m})}, \quad t > t_{r:m}, \quad (35.12)$$

where $F_{T,1}(\cdot)$ and $F_{T,2}(\cdot)$ are as described in Proposition 2.1. If the denominator is extremely close to 0, for example, $< 10^{-8}$, we then conclude the estimates diverge.

As Ye and Ng [27] mentioned, the experimenter should choose the values of N and N_1 wisely. In particular, N should be sufficiently large so that the final estimates obtained from the St-EM algorithm closely resemble the MLEs of the model parameters. In the simulation, we choose $N_1 = 304$ and $N = 800$.

$$U_{j:m} = \begin{cases} \ln \left(\frac{T_{j:m}}{\theta_1} \right), & T_{j:m} < \tau, \\ \ln \left(\frac{T_{j:m} - \tau}{\theta_2} + \frac{\tau}{\theta_1} \right), & T_{j:m} > \tau, \end{cases} \quad (35.13)$$

and denote $\mathbf{U}_1 = (U_{1:m}, \dots, U_{n_1:m})'$ and $\mathbf{U}_2 = (U_{n_1+1:m}, \dots, U_{r:m})'$. By neglecting the fact that all entries in \mathbf{U}_1 are less than $\ln \tau/\theta_1$, we treat \mathbf{U}_1 as an ordinary Type-II censored sample from extreme-value distribution with location parameter $\mu = \ln \theta_1$ and scale parameter $\sigma = \frac{1}{\beta}$, and do similarly for \mathbf{U}_2 . Let us further denote $\boldsymbol{\mu}_1 = (\mu_{1:m}, \dots, \mu_{n_1:m})'$, $\boldsymbol{\mu}_2 = (\mu_{n_1+1:m}, \dots, \mu_{r:m})'$, $\boldsymbol{\Sigma}_1 = ((\sigma_{i,j:m}))$ for $i, j = 1, \dots, n_1$, $\boldsymbol{\Sigma}_2 = ((\sigma_{i,j:m}))$ for $i, j = n_1 + 1, \dots, r$, where $\mu_{j:m} = E(U_{j:m})$ and $\sigma_{i,j:m} = \text{cov}(U_{i:m}, U_{j:m})$. Balakrishnan et al. [18] derived the following expressions for n -component extreme-value distributed coherent system lifetimes:

35.3.3 Providing Initial Values to the Algorithm

In this subsection, we propose initial values by using best linear unbiased estimators (BLUEs) and moment estimates under the condition $1 \leq n_1 \leq r - 1$. Now, let

$$\begin{aligned} \mu_{1:m} &= m \sum_{\substack{j_1, \dots, j_n \geq 0 \\ j_1 + \dots + j_n = m-1}} \binom{m-1}{j_1, \dots, j_n} a_1^{j_1} \dots a_n^{j_n} \\ &\times \sum_{i=1}^n \frac{a_i i [-\gamma - \ln(j_1 + 2j_2 + \dots + nj_n + i)]}{(j_1 + 2j_2 + \dots + nj_n + i)}, \end{aligned} \quad (35.14)$$

$$\begin{aligned} \mu_{1:m}^{(2)} &= m \sum_{\substack{j_1, \dots, j_n \geq 0 \\ j_1 + \dots + j_n = m-1}} \binom{m-1}{j_1, \dots, j_n} a_1^{j_1} \dots a_n^{j_n} \\ &\times \sum_{i=1}^n \frac{a_i i \left\{ \frac{\pi^2}{6} + [\gamma + \ln(j_1 + 2j_2 + \dots + nj_n + i)]^2 \right\}}{(j_1 + 2j_2 + \dots + nj_n + i)}, \end{aligned} \quad (35.15)$$

$$\begin{aligned} \mu_{s,s+1:m} &= \frac{m!}{(s-1)!(m-s-1)!} \sum_{l=0}^{s-1} (-1)^l \binom{s-1}{l} \sum_{\substack{k_1, \dots, k_n \geq 0 \\ k_1 + \dots + k_n = l}} \binom{l}{k_1, \dots, k_n} \\ &\times \sum_{\substack{j_1, \dots, j_n \geq 0 \\ j_1 + \dots + j_n = m-s-1}} \binom{m-s-1}{j_1, \dots, j_n} a_1^{j_1+k_1} \dots a_n^{j_n+k_n} \sum_{i=1}^n \sum_{i'=1}^n a_i a_{i'} i i' \phi(\delta, \alpha), \end{aligned} \quad (35.16)$$

where $\mu_{j:m}^{(2)} = E[(U_{j:m})^2]$, $\mu_{i,j:m} = E(U_{i:m} U_{j:m}^*)$, $\gamma = 0.57721566490 \dots$ is the Euler-Mascheroni constant, $\delta = k_1 + 2k_2 + \dots + nk_n + i$, $\alpha = j_1 + 2j_2 + \dots + nj_n + i'$, and

We can then find all the values of $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$, $\boldsymbol{\Sigma}_1$, and $\boldsymbol{\Sigma}_2$, by further using the well-known triangle and rectangle rules for moments of order statistics (see Arnold et al. [26]), as

$$\begin{aligned} \phi(\delta, \alpha) &= \int_{-\infty}^{\infty} \int_{-\infty}^u t u e^{t-\delta e^t} e^{u-\alpha e^u} dt du, \\ s\mu_{s+1:m}^{(l)} + (m-s)\mu_{s:m}^{(l)} &= m\mu_{s:m-1}^{(l)}, \\ s &= 1, 2, \dots, m-1, l = 1, 2, \end{aligned} \quad (35.17)$$

$$\begin{aligned}
 (i-1)\mu_{i,j:m} + (j-i)\mu_{i-1,j:m} + (m-j+1)\mu_{i-1,j-1:m} \\
 = m\mu_{i-1,j-1:m-1}, \\
 2 \leq i < j \leq m. \quad (35.18)
 \end{aligned}$$

By using the transferred data $U_1 = (U_{1:m}, \dots, U_{n_1:m})$ obtained from the first stress level, we then have the BLUEs of μ and σ as

$$\tilde{\mu} = \mu'_1 \Gamma U_1 \quad \text{and} \quad \tilde{\sigma} = -\mathbf{1}' \Gamma U_1, \quad (35.19)$$

where

$$\Gamma = \frac{\Sigma_1^{-1} (\mu_1 \mathbf{1}' - \mathbf{1} \mu'_1) \Sigma_1^{-1}}{(\mu'_1 \Sigma_1^{-1} \mu_1) (\mathbf{1}' \Sigma_1^{-1} \mathbf{1}) - (\mu'_1 \Sigma_1^{-1} \mathbf{1})^2}.$$

Using these, we obtain the initial values of θ_1 and β as

$$\tilde{\theta}_1 = e^{\tilde{\mu}} \quad \text{and} \quad \tilde{\beta} = \tilde{\sigma}^{-1}. \quad (35.20)$$

At the second stress level, we replace θ_1, β by $\tilde{\theta}_1$ and $\tilde{\beta}$. Now, by equating

$$\begin{aligned}
 \sum_{i=n_1+1}^r (m-i+1) \left(\frac{T_{i:m} - \tau}{\theta_2} + \frac{\tau}{\tilde{\theta}_1} \right) \\
 = \sum_{i=n_1+1}^r (m-i+1) e^{\frac{\mu_{i:m}}{\tilde{\beta}}}, \quad (35.21)
 \end{aligned}$$

we then obtain the estimate of θ_2 as

$$\tilde{\theta}_2 = \frac{\sum_{i=n_1+1}^r (m-i+1)(T_{i:m} - \tau)}{\sum_{i=n_1+1}^r (m-i+1) \left(e^{\frac{\mu_{i:m}}{\tilde{\beta}}} - \frac{\tau}{\tilde{\theta}_1} \right)}. \quad (35.22)$$

To provide the initial values in the case when $n_1=1$, one can use the following two equations

$$\begin{aligned}
 \ln T_{1:m} = \frac{1}{\tilde{\beta}} \mu_{1:m} + \ln \theta_1 \quad \text{and} \\
 S_T(\tau) = \frac{m}{m+1}, \quad (35.23)
 \end{aligned}$$

where $S_T(\cdot)$ is as defined before and then use the above procedure to obtain the estimate of θ_2 . When $n_1 = 0$, one can simply take $\tilde{\theta}_1 = \tau$ and $\tilde{\beta} = 1$ and, then by using them, can proceed to obtain $\tilde{\theta}_2$ as described above.

35.3.4 Confidence Intervals

Observed Fisher Information Matrix

From the asymptotic properties of MLEs, it is known that as $m \rightarrow \infty$, we have

$$\sqrt{m}(\hat{\theta} - \theta) \xrightarrow{D} N(\mathbf{0}, \mathbf{I}^{-1}(\theta)),$$

where $\mathbf{I}(\theta)$ is estimated by the observed information matrix. Then, one can easily construct asymptotic CIs for the model parameters. The observed information matrix is given by

$$I_{obs}(\theta) = \begin{pmatrix} -\frac{\partial^2 \log L}{\partial \theta_1^2} & -\frac{\partial^2 \log L}{\partial \theta_1 \partial \theta_2} & -\frac{\partial^2 \log L}{\partial \theta_1 \partial \beta} \\ -\frac{\partial^2 \log L}{\partial \theta_2 \partial \theta_1} & -\frac{\partial^2 \log L}{\partial \theta_2^2} & -\frac{\partial^2 \log L}{\partial \theta_2 \partial \beta} \\ -\frac{\partial^2 \log L}{\partial \beta \partial \theta_1} & -\frac{\partial^2 \log L}{\partial \beta \partial \theta_2} & -\frac{\partial^2 \log L}{\partial \beta^2} \end{pmatrix}_{\theta=\hat{\theta}}.$$

The elements of the observed information matrix are presented in the Appendix.

Missing Information Principle

Louis's [32] missing information principle is a commonly used technique to obtain asymptotic variance-covariance matrix of the MLEs within the EM algorithm framework. Ye and Ng [27] adopted the missing information principle successfully within the framework of the St-EM algorithm. We follow Ye and Ng's [27] approach to construct the asymptotic variance-covariance matrix of the parameter estimates obtained by the St-EM algorithm.

Let the pseudo-complete data be denoted by

$$\mathbf{t}_{pc} = \mathbf{t} \cup \mathbf{t}^{(g)},$$

where $\mathbf{t} = t_{1:m} < \dots < t_{r:m}$ is the observed data and $\mathbf{t}^{(g)} = t_{r+1:m}^g < \dots < t_{m:m}^g$ is the randomly generated data from the appropriate conditional distributions in (35.10), (35.11), and (35.12). Let $H_1(\mathbf{t}_{pc}, \theta)$ and $H_2(\mathbf{t}_{pc}, \theta)$ denote the matrix of first derivatives and matrix of the negative of the second derivatives of the pseudo-complete likelihood with respect to θ . Then, by using the missing information principle following Ye and Ng [27], the observed information matrix is given by

$$\mathbf{I}(\theta) = E[H_2(\mathbf{t}_{pc}, \theta)|\mathbf{t}] - E[H_1^2(\mathbf{t}_{pc}, \theta)|\mathbf{t}] + \{E[H_1(\mathbf{t}_{pc}, \theta)|\mathbf{t}]\}^2. \quad (35.24)$$

For evaluating (35.24), we impute R samples $\mathbf{t}^{(g)(j)}, j = 1, \dots, R$ corresponding to the missing data, conditional on the observed data \mathbf{t} and θ , by generating samples from the appropriate conditional distributions (35.10), (35.11), and (35.12), and obtain the pseudo-complete data $\mathbf{t}_{pc}^{(j)}, j = 1, \dots, R$. Finally, estimate of (35.24) is obtained as

$$\begin{aligned}
 \widehat{\mathbf{I}}(\hat{\theta}) = \frac{1}{R} \sum_{j=1}^R H_2(\mathbf{t}_{pc}^{(j)}, \theta) - \frac{1}{R} \sum_{j=1}^R [H_1(\mathbf{t}_{pc}^{(j)}, \theta)]^2 \\
 + \left[\frac{1}{R} \sum_{j=1}^R H_1(\mathbf{t}_{pc}^{(j)}, \theta) \right]^2. \quad (35.25)
 \end{aligned}$$

Finally, (35.25) is evaluated at the estimated value of θ obtained from the St-EM algorithm, and the resulting matrix is then inverted to obtain the asymptotic variance-covariance matrix of the parameter estimates. The asymptotic confidence intervals for the model parameters can then be easily constructed.

Bootstrap Confidence Intervals

It is also possible to construct parametric bootstrap CIs for the parameters. To do so, we first compute the MLE $\hat{\theta}$ based on the Type-II censored data on $m n$ -component systems. Then, treating $\hat{\theta}$ as the true value of the parameter θ , in the same sampling framework, we generate a sample of n -component systems of size m and calculate the MLE $\hat{\theta}^*$ for this sample. This process is repeated B times, to get B bootstrap estimates $\hat{\theta}_{(1)}^*, \dots, \hat{\theta}_{(B)}^*$. Finally, a $100(1 - \alpha)\%$ CI for θ_1 , for example, can be obtained as

$$\text{LCL: } \hat{\theta}_1 - b_{\theta_1} - z_{\alpha/2}\sqrt{v_{\theta_1}}, \quad \text{UCL: } \hat{\theta}_1 - b_{\theta_1} + z_{\alpha/2}\sqrt{v_{\theta_1}},$$

where b_{θ_1} and v_{θ_1} are the bootstrap bias and variance, respectively, for the estimate of θ_1 , calculated based on the B bootstrap estimates, and z_{α} is the upper α -percentage point of a standard normal variable. The same method applies to the other parameters θ_2 and β as well.

One more variant of the $100(1 - \alpha)\%$ bootstrap CIs would be

$$\hat{\theta}_{1((\frac{\alpha B}{2})^*)}^*, \hat{\theta}_{1((1 - \frac{\alpha}{2})B)^*}^*,$$

where $\hat{\theta}_{1(k)}^*$ is the k -th smallest value in the ordered sequence of B bootstrap estimates for θ_1 , $[x]$ representing the greatest integer contained in x . One can refer to Efron and Tibshirani [33] for elaborate details on various bootstrap CIs.

Note that the inferential procedures described above are carried out conditionally on the number of failures between 1 and $r - 1$, i.e., $1 \leq n_1 \leq r - 1$, since this is the condition for the existence of the MLEs. Naturally, the different confidence intervals can be obtained under this condition only, and all subsequent computational results are carried out conditionally on this.

35.4 Outline of Bayesian Analysis

Many researchers have explored Bayesian analysis of reliability data [34]. In particular, Ganguly et al. [35] discussed Bayesian analysis of data from simple step-stress experiments under the assumption of Weibull lifetimes. Recently, Jablonka et al. [36] have presented a discussion on Bayesian inference for coherent systems with Weibull distributed components. Following these works, we present here an outline of Bayesian inference for a simple step-stress experiment

involving coherent systems with Weibull distributed components with known signatures.

While prior elicitation is an involved problem in itself, here, following Berger and Sun [37], Kundu [38], and Ganguly et al. [35], we assume gamma priors on the scale parameters θ_1 and θ_2 , i.e.,

$$\theta_1 \sim \pi_1(\theta_1), \quad \theta_2 \sim \pi_2(\theta_2),$$

where

$$\pi_i(\theta_i) \propto \theta_i^{a_i-1} e^{-b_i\theta_i}, \quad \theta_i > 0, \quad (35.26)$$

with a_i and b_i as known hyper-parameters, $i = 1, 2$. Further, we assume a gamma prior on the shape parameter β , i.e., for known hyper-parameters a_3 and b_3

$$\beta \sim \pi_3(\beta), \quad (35.27)$$

where

$$\pi_3(\beta) \propto \beta^{a_3-1} e^{-b_3\beta}, \quad \beta > 0, a_3, b_3 > 0. \quad (35.28)$$

It may be noted here that for building the theory, it is also possible to just assume that the prior on β is a log-concave density, without referring to any specific form for it [36]. Finally, we assume that the priors are independent.

The joint posterior distribution of θ_1 , θ_2 , and β is then given by

$$\begin{aligned} p(\theta_1, \theta_2, \beta | DATA) &\propto L(\theta) \times \pi_1(\theta_1) \times \pi_2(\theta_2) \times \pi_3(\beta) \\ &\propto L(\theta) \times \theta_1^{a_1-1} \theta_2^{a_2-1} \\ &\quad \beta^{a_3-1} e^{-(b_1\theta_1 + b_2\theta_2 + b_3\beta)}, \end{aligned} \quad (35.29)$$

with $L(\theta)$ as the likelihood function as given in Sect. 35.3.1 for different values of n_1 . Thus, under squared error loss function, the Bayes estimates of any parametric function $p(\theta_1, \theta_2, \beta)$ are given by

$$\begin{aligned} p(\widehat{\theta_1}, \widehat{\theta_2}, \widehat{\beta}) &= \int_0^\infty \int_0^\infty \int_0^\infty K \times p(\theta_1, \theta_2, \beta) L(\theta) \theta_1^{a_1-1} \theta_2^{a_2-1} \\ &\quad \beta^{a_3-1} e^{-(b_1\theta_1 + b_2\theta_2 + b_3\beta)} d\theta_1 d\theta_2 d\beta, \end{aligned} \quad (35.30)$$

K being a normalizing constant. Note that it is not possible to obtain closed form expression for this Bayes estimate. One can use a general purpose approximation method such as Lindley's approximation [39]. However, unfortunately, credible intervals cannot be constructed in conjunction with Lindley's approximation. Alternatively, one can use an MCMC-based approach to calculate the Bayes estimates and credible intervals. Here, following Ganguly et al. [35], we outline an algorithm based on importance sampling for obtaining

the Bayes estimate and credible intervals for the parametric function $p(\theta_1, \theta_2, \beta)$.

Note that the joint posterior density can be decomposed as

$$\pi(\theta_1, \theta_2, \beta|DATA) = g_1(\theta_1|\beta, DATA) \times g_2(\theta_2|\beta, DATA) \\ \times g_3(\beta|DATA),$$

where forms of $g_1(\cdot)$, $g_2(\cdot)$, $g_3(\cdot)$ will depend upon $L(\theta)$ based on different values of n_1 , i.e., for Cases (i)–(iii) in Sect. 35.3. It is usually possible to approximate $g_3(\beta|DATA)$ by a known log-concave density [35, 38]. To obtain an approximating density, one may plot $g_3(\beta|DATA)$ and then select an appropriate approximating density by equating the mean and variance of $g_3(\beta|DATA)$ to those of the approximating density along the lines of Kundu [38] and Ganguly et al. [35]. Let $g_4(\beta|DATA)$ denote the approximating density. Then, the joint posterior density can be written as

$$\pi(\theta_1, \theta_2, \beta|DATA) = w(\beta) \times g_1(\theta_1|\beta, DATA) \\ \times g_2(\theta_2|\beta, DATA) \times g_4(\beta|DATA),$$

where $w(\beta) = \frac{g_3(\beta|DATA)}{g_4(\beta|DATA)}$ is the weight function. Finally, an algorithm similar to the one proposed by Ganguly et al. [35] may be given as:

Algorithm:

1. Generate β_1 from $g_4(\beta|DATA)$
 2. Given β_1 , generate θ_{11} from $g_1(\theta_1|DATA)$
 3. Given β_1 , generate θ_{21} from $g_2(\theta_2|DATA)$
 4. Repeat steps 1 to 3 N times, to get $(\theta_{11}, \theta_{21}, \beta_1), \dots, (\theta_{1N}, \theta_{2N}, \beta_N)$
 5. Compute $p_i = p(\theta_{1i}, \theta_{2i}, \beta_i), i = 1, \dots, N$
 6. Compute $w_i = w(\beta_i), i = 1, \dots, N$
 7. Compute the Bayes estimate $p(\hat{\theta}_1, \hat{\theta}_2, \hat{\beta}) = \frac{1}{N} \sum_{i=1}^N w_i p_i$.
-

Embedded in this process, credible intervals for $p(\theta_1, \theta_2, \beta)$ can also be constructed using the values $p_i, i = 1, \dots, N$.

35.5 Simulation Study

In this section, we carry out a detailed Monte Carlo simulation study for several three-component systems, with $m = 50$ and 100, and by taking $\theta_1 = 1$ without loss of any generality, to numerically investigate the proposed methods of inference. The values of θ_2 are chosen as 0.50 and 0.25; values of β are chosen as 1.00, 2.00, and 3.00; and values of τ are chosen so that $F_T(\tau) = 0.20$ and 0.30, respectively, for system tests.

Based on these chosen values of the parameters, we compare the performances of the two methods of estimation, namely, the direct likelihood-based approach using numer-

ical optimization through the Newton-Raphson method and the approach based on St-EM algorithm. The results of this comparison are presented in Table 35.2, suggesting that the performances of these two approaches are quite close. For this reason, we will present only the results obtained from the direct likelihood-based approach for the remaining tables.

Based on the above settings, and for m as 150 and 300, we also carry out a simulation study for individual components, the results of which are presented in Tables 35.3 and 35.4, which are then compared with the inference obtained from the system lifetimes. For example, when $F_T(\tau) = 0.20$ for a parallel system, with the same value of τ , we found $F_X(\tau) = 0.58$ for component lifetimes. We compute the means and mean squared errors (MSEs) of $\hat{\theta}_1, \hat{\theta}_2$ and $\hat{\beta}$ through simulations. The coverage probabilities are then equi-tailed CIs obtained by the asymptotic approach. In order to compare the relative efficiency between different coherent systems, we calculate the trace and determinant of the observed Fisher information matrix, as well as the total time on test (TTT). We observe that in almost all system tests, all the MLEs are positively biased.

We would like to point out some relative advantages of testing coherent systems over individual components in this setting here. Note that one would expect to have smaller TTT when testing coherent systems as compared to testing individual components. We observe in Tables 35.3 and 35.4 that this is always the case. This is a clear advantage in the setting of step-stress experiments when TTT is a concern. Also, one may consider the information contained in coherent systems and individual components and compare them. For example, one may compare the information contained in 30 coherent systems with signature $(1/3, 2/3, 0)$ with 50 individual components, as the number of components expected to fail when 30 coherent systems with signature $(1/3, 2/3, 0)$ are tested is 50. For series and parallel systems, the comparison is clear. Note that for three-component series systems, we compare the information obtained from the first failure (or equivalently, the smallest order statistics) among a sample of size 3, with one individual component (since the expected number of failed component of a series system is 1). Clearly, the system will involve more information in this case. However, for three-component parallel systems, we actually compare the information obtained from the third failure (or equivalently, the largest order statistics) among a sample of size 3, with three individual components (since the expected number of failed component of a parallel system is 3). Clearly, the system will involve less information in this case. Thus, under this setting, from the perspective of information contained in failures, testing three-component series systems is preferred over individual components.

Based on the trace and determinant of the observed Fisher information matrix, denoted by $tr(\mathbf{I})$ and $det(\mathbf{I})$, respectively, we find that the parallel system always has the highest

Table 35.2 Simulated values of means and MSEs of the estimates from direct likelihood (in the first line) and St-EM approach (in the second line)

m	r	$F_T(\tau)$	θ_2	$E(\hat{\theta}_1)$	$E(\hat{\theta}_2)$	$E(\hat{\beta})$	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	Prob. of con. (includ. $n_1 = r$)	
$P = (0, 0, 1)$ (equivalently, $\alpha = (3, -3, 1)$)											
$\beta = 1.00$											
50	40	0.20	0.50	1.0407	0.5742	1.1020	0.0514	0.0723	0.1177	0.9929	
				1.0380	0.5862	1.1281	0.0473	0.0739	0.1249	0.9993	
			1.0397	0.2876	1.1034	0.0504	0.0181	0.1186	0.9838		
		0.30	0.50	1.0381	0.2933	1.1289	0.0473	0.0186	0.1268	0.9997	
				1.0265	0.5551	1.0696	0.0234	0.0528	0.0696	0.9972	
			1.0274	0.5610	1.0848	0.0229	0.0534	0.0722	1.0000		
100	80	0.20	0.50	1.0255	0.2782	1.0708	0.0230	0.0132	0.0693	0.9879	
				1.0274	0.2805	1.0848	0.0229	0.0134	0.0722	1.0000	
			1.0179	0.5354	1.0475	0.0180	0.0280	0.0456	0.9906		
		0.30	0.50	1.0168	0.5422	1.0615	0.0174	0.0284	0.0472	1.0000	
				1.0159	0.2685	1.0493	0.0175	0.0070	0.0453	0.9810	
			1.0168	0.2711	1.0615	0.0174	0.0071	0.0472	1.0000		
100	80	0.20	0.50	1.0117	0.5282	1.0338	0.0095	0.0236	0.0302	0.9908	
				1.0123	0.5313	1.0416	0.0094	0.0237	0.0308	1.0000	
			1.0112	0.2645	1.0347	0.0094	0.0059	0.0299	0.9864		
		0.30	0.50	1.0123	0.2656	1.0416	0.0094	0.0059	0.0308	1.0000	
			$\beta = 2.00$								
50	40	0.20	0.50	1.0202	0.5601	2.2080	0.0139	0.0590	0.5685	0.9867	
				1.0205	0.5634	2.2375	0.0131	0.0584	0.5767	0.9995	
			1.0159	0.2827	2.2237	0.0122	0.0145	0.5584	0.9624		
		0.30	0.50	1.0205	0.2819	2.2394	0.0131	0.0149	0.5976	0.9998	
				1.0113	0.5400	2.1327	0.0053	0.0385	0.2926	0.9890	
			1.0128	0.5405	2.1497	0.0054	0.0382	0.2977	1.0000		
100	80	0.20	0.50	1.0104	0.2709	2.1377	0.0052	0.0096	0.2911	0.9721	
				1.0128	0.2702	2.1496	0.0054	0.0096	0.2977	1.0000	
			1.0083	0.5294	2.0978	0.0043	0.0238	0.2167	0.9786		
		0.30	0.50	1.0085	0.5311	2.1125	0.0043	0.0237	0.2193	1.0000	
				1.0052	0.2683	2.1172	0.0040	0.0059	0.2175	0.9371	
			1.0085	0.2655	2.1125	0.0043	0.0059	0.2193	1.0000		
100	80	0.20	0.50	1.0048	0.5205	2.0637	0.0022	0.0177	0.1271	0.9805	
				1.0056	0.5212	2.0735	0.0022	0.0176	0.1286	1.0000	
			1.0038	0.2621	2.0733	0.0022	0.0043	0.1253	0.9601		
		0.30	0.50	1.0056	0.2606	2.0735	0.0022	0.0044	0.1286	1.0000	
			$\beta = 3.00$								
50	40	0.20	0.50	1.0121	0.5555	3.3132	0.0058	0.0533	1.3254	0.9828	
				1.0135	0.5559	3.3417	0.0059	0.0531	1.3466	0.9996	
			1.0095	0.2807	3.3406	0.0053	0.0135	1.3617	0.9561		
		0.30	0.50	1.0135	0.2782	3.3447	0.0059	0.0136	1.3973	0.9998	
				1.0072	0.5351	3.1941	0.0023	0.0346	0.6640	0.9879	
			1.0083	0.5339	3.2116	0.0023	0.0339	0.6696	1.0000		
100	80	0.20	0.50	1.0063	0.2686	3.2017	0.0022	0.0085	0.6565	0.9714	
				1.0083	0.2669	3.2116	0.0023	0.0085	0.6696	1.0000	
			1.0050	0.5268	3.1455	0.0018	0.0217	0.4994	0.9748		
		0.30	0.50	1.0053	0.5276	3.1622	0.0018	0.0218	0.5072	1.0000	
				1.0023	0.2681	3.1843	0.0016	0.0055	0.5056	0.9150	
			1.0053	0.2638	3.1621	0.0018	0.0054	0.5072	1.0000		
100	80	0.20	0.50	1.0031	0.5181	3.0933	0.0010	0.0159	0.2879	0.9790	
				1.0036	0.5179	3.1039	0.0010	0.0158	0.2905	1.0000	
			1.0024	0.2611	3.1082	0.0009	0.0039	0.2870	0.9537		
		0.30	0.50	1.0036	0.2590	3.1039	0.0010	0.0040	0.2905	1.0000	

(continued)

Table 35.2 (continued)

m	r	$F_T(\tau)$	θ_2	$E(\hat{\theta}_1)$	$E(\hat{\theta}_2)$	$E(\hat{\beta})$	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	Prob. of con. (includ. $n_1 = r$)		
$P = (1, 0, 0)$ (equivalently, $\alpha = (0, 0, 1)$)												
$\beta = 1.00$												
50	40	0.20	0.50	1.2515	0.5051	1.0918	1.9512	0.0120	0.0852	0.9931		
				1.2097	0.5004	1.1340	1.5880	0.0116	0.1012	0.9951		
			0.25	1.2528	0.2526	1.0935	1.9698	0.0030	0.0891	0.9924		
		0.30	0.50	1.2102	0.2502	1.1361	1.6009	0.0029	0.1062	0.9985		
				1.1322	0.5078	1.0809	0.7399	0.0122	0.0721	0.9983		
			0.25	1.1117	0.5043	1.1107	0.7103	0.0119	0.0811	1.0000		
100	80	0.20	0.50	1.1325	0.2540	1.0811	0.7407	0.0031	0.0727	0.9959		
				1.1117	0.2521	1.1107	0.7103	0.0030	0.0811	1.0000		
			0.25	1.0944	0.5023	1.0445	0.3883	0.0059	0.0328	0.9977		
		0.30	0.50	1.0768	0.4996	1.0665	0.3726	0.0058	0.0368	0.9999		
				1.0929	0.2512	1.0445	0.3811	0.0015	0.0324	0.9916		
			0.25	1.0766	0.2498	1.0665	0.3721	0.0014	0.0368	1.0000		
50	40	0.20	0.50	1.0585	0.5043	1.0378	0.2126	0.0059	0.0290	0.9978		
				1.0474	0.5022	1.0534	0.2055	0.0059	0.0310	1.0000		
			0.25	1.0584	0.2521	1.0379	0.2123	0.0015	0.0290	0.9944		
		0.30	0.50	1.0474	0.2511	1.0534	0.2055	0.0015	0.0310	1.0000		
				1.1415	0.5194	2.2039	0.9091	0.0089	0.5062	0.9931		
			0.25	1.1102	0.5198	2.2753	0.6004	0.0090	0.5603	0.9973		
100	80	0.20	0.50	1.1396	0.2600	2.2099	0.9078	0.0023	0.5329	0.9914		
				1.1112	0.2602	2.2808	0.6136	0.0023	0.5930	0.9993		
			0.25	1.0538	0.5160	2.1554	0.1935	0.0107	0.3394	0.9980		
		0.30	0.50	1.0426	0.5145	2.1967	0.1796	0.0105	0.3588	1.0000		
				1.0537	0.2580	2.1550	0.1937	0.0027	0.3355	0.9940		
			0.25	1.0426	0.2572	2.1967	0.1796	0.0026	0.3588	1.0000		
50	40	0.20	0.50	1.0467	0.5100	2.1017	0.1330	0.0038	0.2002	0.9964		
				1.0377	0.5102	2.1365	0.1260	0.0038	0.2102	1.0000		
			0.25	1.0468	0.2550	2.1010	0.1328	0.0010	0.1979	0.9923		
		0.30	0.50	1.0377	0.2551	2.1366	0.1259	0.0010	0.2102	1.0000		
				1.0229	0.5081	2.0721	0.0567	0.0049	0.1362	0.9972		
			0.25	1.0176	0.5072	2.0930	0.0552	0.0049	0.1402	1.0000		
100	80	0.20	0.50	1.0234	0.2539	2.0715	0.0567	0.0012	0.1357	0.9942		
				1.0176	0.2536	2.0930	0.0552	0.0012	0.1402	1.0000		
			0.25	1.0934	0.5262	3.3087	0.5752	0.0152	1.2948	0.9903		
		50	40	0.20	0.50	1.0732	0.5272	3.3964	0.3419	0.0151	1.3994	0.9978
						1.0923	0.2635	3.3151	0.5775	0.0039	1.3489	0.9855
					0.25	1.0733	0.2639	3.4035	0.3389	0.0039	1.4739	0.9993
0.30	0.50			1.0298	0.5196	3.2243	0.0818	0.0146	0.7895	0.9961		
				1.0232	0.5178	3.2729	0.0737	0.0142	0.8237	1.0000		
	0.25			1.0295	0.2599	3.2236	0.0819	0.0036	0.7806	0.9919		
100	80	0.20	0.50	1.0232	0.2589	3.2729	0.0737	0.0036	0.8237	1.0000		
				1.0278	0.5134	3.1523	0.0610	0.0064	0.5003	0.9947		
			0.25	1.0227	0.5139	3.1939	0.0587	0.0064	0.5208	1.0000		
		0.30	0.50	1.0252	0.2573	3.1563	0.0601	0.0016	0.5013	0.9806		
				1.0227	0.2569	3.1939	0.0587	0.0016	0.5208	1.0000		
			0.25	1.0127	0.5096	3.1028	0.0245	0.0066	0.3149	0.9959		
50	40	0.20	0.50	1.0095	0.5088	3.1279	0.0239	0.0066	0.3232	1.0000		
				1.0124	0.2549	3.1034	0.0244	0.0017	0.3157	0.9915		
			0.25	1.0095	0.2544	3.1279	0.0239	0.0016	0.3232	1.0000		

(continued)

Table 35.2 (continued)

m	r	$F_T(\tau)$	θ_2	$E(\hat{\theta}_1)$	$E(\hat{\theta}_2)$	$E(\hat{\beta})$	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	Prob. of con. (includ. $n_1 = r$)
$P = (0, 2/3, 1/3)$ (equivalently, $a = (1, 1, -1)$)										
$\beta = 1.00$										
50	40	0.20	0.50	1.0733	0.5434	1.1028	0.2273	0.0298	0.1101	0.9957
				1.0554	0.5526	1.1385	0.2039	0.0310	0.1224	0.9986
			0.25	1.0738	0.2720	1.1038	0.2296	0.0077	0.1152	0.9919
		0.30	0.50	1.0560	0.2766	1.1396	0.2044	0.0080	0.1278	0.9999
				1.0390	0.5364	1.0734	0.0886	0.0287	0.0687	0.9986
			0.25	1.0315	0.5407	1.0948	0.0828	0.0290	0.0730	1.0000
100	80	0.20	0.50	1.0390	0.2683	1.0735	0.0887	0.0072	0.0688	0.9957
				1.0315	0.2703	1.0948	0.0828	0.0073	0.0730	1.0000
			0.25	1.0346	0.5190	1.0465	0.0734	0.0117	0.0410	0.9977
		0.30	0.50	1.0265	0.5240	1.0651	0.0695	0.0120	0.0439	1.0000
				1.0351	0.2595	1.0460	0.0738	0.0029	0.0410	0.9901
			0.25	1.0265	0.2620	1.0651	0.0695	0.0030	0.0439	1.0000
50	40	0.20	0.50	1.0184	0.5169	1.0348	0.0327	0.0125	0.0288	0.9974
				1.0149	0.5193	1.0461	0.0318	0.0126	0.0300	1.0000
			0.25	1.0177	0.2587	1.0350	0.0327	0.0031	0.0288	0.9899
		0.30	0.50	1.0149	0.2597	1.0461	0.0318	0.0031	0.0300	1.0000
				1.0400	0.5456	2.2126	0.0762	0.0368	0.5640	0.9929
			0.25	1.0337	0.5503	2.2606	0.0655	0.0368	0.5880	0.9992
100	80	0.20	0.50	1.0383	0.2736	2.2183	0.0738	0.0095	0.5911	0.9855
				1.0336	0.2754	2.2636	0.0655	0.0095	0.6159	0.9999
			0.25	1.0172	0.5313	2.1379	0.0212	0.0275	0.2963	0.9959
		0.30	0.50	1.0153	0.5323	2.1652	0.0203	0.0272	0.3055	1.0000
				1.0168	0.2659	2.1392	0.0212	0.0069	0.2964	0.9890
			0.25	1.0153	0.2662	2.1652	0.0203	0.0068	0.3054	1.0000
50	40	0.20	0.50	1.0173	0.5200	2.0953	0.0214	0.0145	0.2088	0.9915
				1.0153	0.5221	2.1186	0.0206	0.0145	0.2136	1.0000
			0.25	1.0155	0.2603	2.0974	0.0206	0.0036	0.2086	0.9794
		0.30	0.50	1.0153	0.2611	2.1185	0.0206	0.0036	0.2136	1.0000
				1.0078	0.5148	2.0659	0.0079	0.0122	0.1253	0.9925
			0.25	1.0072	0.5151	2.0791	0.0079	0.0121	0.1275	1.0000
100	80	0.20	0.50	1.0072	0.2576	2.0671	0.0078	0.0030	0.1249	0.9835
				1.0072	0.2575	2.0791	0.0079	0.0030	0.1275	1.0000
			0.25	1.0249	0.5460	3.3171	0.0368	0.0395	1.3288	0.9894
		0.30	0.50	1.0224	0.5483	3.3729	0.0315	0.0394	1.3936	0.9994
				1.0213	0.2746	3.3363	0.0348	0.0101	1.4102	0.9773
			0.25	1.0224	0.2743	3.3763	0.0313	0.0101	1.4486	0.9999
50	40	0.20	0.50	1.0103	0.5295	3.2016	0.0087	0.0273	0.6749	0.9948
				1.0098	0.5288	3.2310	0.0084	0.0269	0.6889	1.0000
			0.25	1.0091	0.2654	3.2074	0.0082	0.0069	0.6770	0.9865
		0.30	0.50	1.0098	0.2644	3.2310	0.0084	0.0067	0.6889	1.0000
				1.0104	0.5200	3.1418	0.0093	0.0156	0.4936	0.9895
			0.25	1.0097	0.5209	3.1668	0.0091	0.0155	0.5021	1.0000
100	80	0.20	0.50	1.0073	0.2615	3.1567	0.0087	0.0039	0.4941	0.9646
				1.0097	0.2604	3.1668	0.0091	0.0039	0.5020	1.0000
			0.25	1.0048	0.5137	3.0951	0.0034	0.0122	0.2853	0.9944
		0.30	0.50	1.0046	0.5134	3.1102	0.0034	0.0121	0.2888	1.0000
				1.0041	0.2575	3.0996	0.0033	0.0030	0.2852	0.9816
			0.25	1.0046	0.2567	3.1102	0.0034	0.0030	0.2888	1.0000

(continued)

Table 35.2 (continued)

m	r	$F_T(\tau)$	θ_2	$E(\hat{\theta}_1)$	$E(\hat{\theta}_2)$	$E(\hat{\beta})$	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	Prob. of con. (includ. $n_1 = r$)
$P = (1/3, 2/3, 0)$ (equivalently, $\alpha = (0, 2, -1)$)										
$\beta = 1.00$										
50	40	0.20	0.50	1.1631	0.5180	1.0984	0.7661	0.0106	0.0993	0.9923
				1.1336	0.5214	1.1394	0.6994	0.0110	0.1157	0.9958
			0.25	1.1627	0.2592	1.0996	0.7668	0.0027	0.1016	0.9904
		0.30	0.50	1.1345	0.2608	1.1407	0.7169	0.0028	0.1176	0.9984
				1.0752	0.5205	1.0858	0.2840	0.0139	0.0819	0.9986
			0.25	1.0599	0.5223	1.1139	0.2651	0.0141	0.0897	0.9999
100	80	0.20	0.50	1.0747	0.2603	1.0861	0.2835	0.0035	0.0824	0.9960
				1.0597	0.2612	1.1140	0.2633	0.0035	0.0902	1.0000
			0.25	1.0716	0.5085	1.0456	0.2082	0.0047	0.0381	0.9974
		0.30	0.50	1.0574	0.5104	1.0669	0.1969	0.0048	0.0419	1.0000
				1.0723	0.2543	1.0455	0.2087	0.0012	0.0381	0.9941
			0.25	1.0574	0.2552	1.0669	0.1969	0.0012	0.0419	1.0000
100	80	0.30	0.50	1.0351	0.5101	1.0402	0.0989	0.0062	0.0328	0.9980
				1.0280	0.5111	1.0549	0.0959	0.0063	0.0348	1.0000
			0.25	1.0355	0.2551	1.0399	0.0992	0.0016	0.0328	0.9929
		0.30	0.50	1.0280	0.2555	1.0549	0.0959	0.0016	0.0348	1.0000
				1.0723	0.2543	1.0455	0.2087	0.0012	0.0381	0.9941
			0.25	1.0574	0.2552	1.0669	0.1969	0.0012	0.0419	1.0000
$\beta = 2.00$										
50	40	0.20	0.50	1.0898	0.5307	2.2155	0.2826	0.0199	0.5600	0.9910
				1.0761	0.5347	2.2815	0.2489	0.0202	0.6200	0.9974
			0.25	1.0898	0.2658	2.2193	0.2841	0.0051	0.5876	0.9858
		0.30	0.50	1.0767	0.2677	2.2859	0.2577	0.0052	0.6580	0.9988
				1.0322	0.5252	2.1653	0.0715	0.0191	0.3754	0.9968
			0.25	1.0267	0.5258	2.2043	0.0671	0.0190	0.3954	1.0000
100	80	0.20	0.50	1.0321	0.2627	2.1655	0.0716	0.0048	0.3763	0.9924
				1.0249	0.2610	2.1953	0.0572	0.0045	0.3742	1.0000
			0.25	1.0368	0.5143	2.0986	0.0686	0.0084	0.2177	0.9963
		0.30	0.50	1.0320	0.5162	2.1302	0.0654	0.0084	0.2267	1.0000
				1.0358	0.2574	2.0996	0.0684	0.0021	0.2188	0.9859
			0.25	1.0320	0.2581	2.1302	0.0654	0.0021	0.2267	1.0000
100	80	0.30	0.50	1.0152	0.5121	2.0764	0.0255	0.0085	0.1509	0.9972
				1.0129	0.5124	2.0955	0.0251	0.0084	0.1554	1.0000
			0.25	1.0148	0.2562	2.0767	0.0253	0.0021	0.1512	0.9906
		0.30	0.50	1.0129	0.2562	2.0954	0.0251	0.0021	0.1554	1.0000
				1.0320	0.5162	2.1302	0.0654	0.0084	0.2267	1.0000
			0.25	1.0358	0.2574	2.0996	0.0684	0.0021	0.2188	0.9859
$\beta = 3.00$										
50	40	0.20	0.50	1.0585	0.5349	3.3230	0.1505	0.0265	1.3693	0.9872
				1.0515	0.5381	3.4069	0.1307	0.0270	1.5155	0.9979
			0.25	1.0516	0.2691	3.3432	0.1281	0.0069	1.4711	0.9736
		0.30	0.50	1.0519	0.2694	3.4136	0.1355	0.0070	1.6186	0.9988
				1.0186	0.5266	3.2407	0.0299	0.0225	0.8649	0.9951
			0.25	1.0161	0.5260	3.2868	0.0279	0.0221	0.9042	1.0000
100	80	0.20	0.50	1.0175	0.2636	3.2435	0.0293	0.0056	0.8561	0.9881
				1.0161	0.2630	3.2868	0.0279	0.0055	0.9042	1.0000
			0.25	1.0225	0.5159	3.1459	0.0312	0.0111	0.5250	0.9930
		0.30	0.50	1.0200	0.5174	3.1855	0.0303	0.0112	0.5538	1.0000
				1.0158	0.2600	3.1663	0.0290	0.0028	0.5427	0.9554
			0.25	1.0200	0.2587	3.1855	0.0303	0.0028	0.5537	1.0000
100	80	0.30	0.50	1.0088	0.5127	3.1110	0.0108	0.0100	0.3501	0.9946
				1.0079	0.5123	3.1325	0.0107	0.0099	0.3570	1.0000
			0.25	1.0070	0.2570	3.1158	0.0104	0.0025	0.3497	0.9819
		0.30	0.50	1.0079	0.2570	3.1158	0.0104	0.0025	0.3497	0.9819
				1.0079	0.2562	3.1325	0.0107	0.0025	0.3570	1.0000
			0.25	1.0079	0.2562	3.1325	0.0107	0.0025	0.3570	1.0000

Table 35.3 Simulated values of means and MSEs of the MLEs, coverage probabilities of 95% some three-component coherent systems and component life tests. The quantity $F(\tau)$ in the third equi-tailed CIs from observed Fisher information matrix and bootstrap method with $\theta_1 = 1.00$, column corresponds to $F_T(\tau)$ in case of system lifetimes and corresponds to $F_X(\tau)$ in case of TTT, trace and determinant of observed Fisher information matrix, and Type-I error/power for component lifetimes

m	r	$F(\tau)$	θ_2	$E(\hat{\theta}_1)$	$E(\hat{\theta}_2)$	$E(\hat{\beta})$	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	CP_{θ_1}	CP_{θ_2}	CP_{β}	$tr(\mathbf{I})$	$det(\mathbf{I})$	TTT	$\frac{Error}{Power_{TL}}$	$\frac{Error}{Powers}$	
$\mathbf{P} = (0, 0, 1)$ (equivalently, $\mathbf{a} = (3, -3, 1)$)																		
$\beta = 1.00$																		
50	40	0.20	0.50	1.0407	0.5742	1.1020	0.0514	0.0723	0.1177	0.9316	0.9368	0.9462	374.562	130021	61.1718	0.0534	0.0538	
		0.25	0.25	1.0397	0.2876	1.1034	0.0504	0.0181	0.1186	0.9313	0.9403	0.9485	952.775	515840	51.1495	0.0521	0.0515	
	0.30	0.50	1.0265	0.5551	1.0696	0.0234	0.0528	0.0696	0.0696	0.9615	0.9432	0.9526	336.596	186676	65.4578	0.0524	0.0474	
		0.25	0.25	1.0255	0.2782	1.0708	0.0230	0.0132	0.0693	0.9616	0.9473	0.9547	795.392	738470	57.5685	0.0510	0.0453	
100	80	0.20	0.50	1.0179	0.5354	1.0475	0.0180	0.0280	0.0456	0.9443	0.9417	0.9462	679.724	962583	122.3901	0.0562	0.0538	
		0.25	0.25	1.0159	0.2685	1.0493	0.0175	0.0070	0.0453	0.9428	0.9467	0.9490	1738.574	3829992	102.3079	0.0546	0.0510	
	0.30	0.50	1.0117	0.5282	1.0338	0.0095	0.0236	0.0302	0.0302	0.9607	0.9432	0.9469	620.347	1364570	130.9355	0.0556	0.0531	
		0.25	0.25	1.0112	0.2645	1.0347	0.0094	0.0059	0.0299	0.9606	0.9480	0.9488	1451.038	5418426	115.1361	0.0539	0.0512	
Individual components																		
150	120	0.58	0.50	1.0070	0.5054	1.0110	0.0134	0.0117	0.0097	0.9498	0.9368	0.9549	401.809	1335690	103.7597	0.0459	0.0451	
		0.25	0.25	1.0076	0.2530	1.0110	0.0135	0.0029	0.0097	0.9496	0.9377	0.9551	830.845	5323140	95.7502	0.0457	0.0449	
	0.67	0.50	1.0048	0.5041	1.0091	0.0108	0.0187	0.0082	0.0082	0.9480	0.9245	0.9542	364.624	1154394	110.0586	0.0459	0.0458	
		0.25	0.25	1.0060	0.2526	1.0088	0.0109	0.0047	0.0082	0.9483	0.9260	0.9545	646.330	4589176	105.2740	0.0457	0.0455	
300	240	0.58	0.50	1.0029	0.5012	1.0049	0.0066	0.0055	0.0046	0.9478	0.9426	0.9552	781.437	10189479	207.4300	0.0458	0.0448	
		0.25	0.25	1.0038	0.2505	1.0046	0.0064	0.0014	0.0047	0.9519	0.9429	0.9532	1602.226	40711433	191.4534	0.0474	0.0468	
	0.67	0.50	1.0008	0.5001	1.0038	0.0052	0.0085	0.0039	0.0039	0.9488	0.9341	0.9530	705.258	8557692	219.9538	0.0482	0.0470	
		0.25	0.25	1.0027	0.2501	1.0036	0.0052	0.0021	0.0039	0.9507	0.9349	0.9506	1221.771	34155756	210.4999	0.0498	0.0494	
50	40	0.20	0.50	1.0202	0.5601	2.2080	0.0139	0.0590	0.5685	0.9407	0.9375	0.9456	748.707	163228	53.3967	0.6857	0.4240	
		0.25	0.25	1.0159	0.2827	2.2237	0.0122	0.0145	0.5584	0.9399	0.9500	0.9534	1667.405	641869	49.2425	0.6995	0.4292	
	0.30	0.50	1.0113	0.5400	2.1327	0.0053	0.0385	0.2926	0.2926	0.9651	0.9431	0.9533	635.664	234530	55.5486	0.8298	0.6834	
		0.25	0.25	1.0104	0.2709	2.1377	0.0052	0.0096	0.2911	0.9645	0.9495	0.9568	1301.088	927368	52.4720	0.8353	0.6858	
100	80	0.20	0.50	1.0083	0.5294	2.0978	0.0043	0.0238	0.2167	0.9502	0.9425	0.9485	1254.693	1253178	106.8247	0.9227	0.8278	
		0.25	0.25	1.0052	0.2683	2.1172	0.0040	0.0059	0.2175	0.9489	0.9518	0.9551	2877.943	4921550	98.5070	0.9350	0.8490	
	0.30	0.50	1.0048	0.5205	2.0637	0.0022	0.0177	0.1271	0.1271	0.9630	0.9457	0.9481	1134.892	1783188	111.1100	0.9830	0.9630	
		0.25	0.25	1.0038	0.2621	2.0733	0.0022	0.0043	0.1253	0.9630	0.9545	0.9525	2301.959	7029681	104.9451	0.9881	0.9703	
Individual components																		
150	120	0.58	0.50	1.0029	0.5033	2.0192	0.0033	0.0103	0.0390	0.9522	0.9369	0.9544	624.141	1667837	115.7361	1.0000	1.0000	
		0.25	0.25	1.0030	0.2516	2.0188	0.0033	0.0026	0.0390	0.9520	0.9368	0.9543	1192.739	6665878	112.0260	1.0000	1.0000	
	0.67	0.50	1.0021	0.5025	2.0162	0.0027	0.0173	0.0326	0.0326	0.9497	0.9235	0.9549	589.041	1351948	118.9416	1.0000	1.0000	
		0.25	0.25	1.0020	0.2513	2.0164	0.0027	0.0043	0.0327	0.9498	0.9234	0.9543	932.515	5400461	116.8253	1.0000	1.0000	
300	240	0.58	0.50	1.0013	0.5015	2.0087	0.0016	0.0050	0.0188	0.9511	0.9412	0.9529	1206.179	12869091	231.4992	1.0000	1.0000	
		0.25	0.25	1.0016	0.2506	2.0086	0.0016	0.0012	0.0187	0.9527	0.9432	0.9529	2274.347	51408470	224.1136	1.0000	1.0000	
	0.67	0.50	1.0009	0.5008	2.0074	0.0013	0.0076	0.0154	0.0154	0.9508	0.9355	0.9550	1138.035	10027028	237.9151	1.0000	1.0000	
		0.25	0.25	1.0014	0.2500	2.0072	0.0013	0.0019	0.0158	0.9497	0.9346	0.9536	1759.279	40227508	233.7402	1.0000	1.0000	

$\beta = 3.00$																	
50	40	0.20	0.50	1.0121	0.5555	3.3132	0.0058	0.0533	1.3254	0.9445	0.9430	0.9483	1407.724	185865	51.8289	0.9397	0.7723
		0.25	1.0095	3.3406	0.0053	0.0135	1.3617	0.9436	0.9518	0.9551	2537.189	730809	49.2129	0.9495	0.7856		
		0.30	0.50	1.0072	0.5351	3.1941	0.0023	0.0346	0.6640	0.9662	0.9449	0.9541	1215.080	262766	53.2706	0.9893	0.9563
		0.25	1.0063	3.2017	0.0022	0.0085	0.6565	0.9657	0.9499	0.9569	1984.139	1039364	51.3723	0.9918	0.9614		
100	80	0.20	0.50	1.0050	0.5268	3.1455	0.0018	0.0217	0.4994	0.9530	0.9463	0.9492	2325.296	1435915	103.6793	0.9971	0.9848
		0.25	1.0023	3.1843	0.0016	0.0055	0.5056	0.9509	0.9555	0.9550	4247.405	5600783	98.4395	0.9974	0.9868		
		0.30	0.50	1.0031	0.5181	3.0933	0.0010	0.0159	0.2879	0.9611	0.9454	0.9479	2172.367	2007312	106.5465	1.0000	0.9997
		0.25	1.0024	3.1082	0.0009	0.0039	0.2870	0.9611	0.9533	0.9522	3514.754	7891845	102.7519	1.0000	0.9997		
Individual components																	
150	120	0.58	0.50	1.0019	0.5025	3.0274	0.0015	0.0100	0.0879	0.9535	0.9364	0.9535	1159.101	1836530	123.4474	1.0000	1.0000
		0.25	1.0018	3.0271	0.0015	0.0025	0.0880	0.9533	0.9356	0.9533	1790.889	7344783	121.0295	1.0000	1.0000		
		0.67	0.50	1.0012	0.5018	3.0235	0.0012	0.0171	0.0736	0.9496	0.9223	0.9543	1143.737	1449877	125.5865	1.0000	1.0000
		0.25	1.0011	3.0238	0.0012	0.0043	0.0732	0.9496	0.9234	0.9547	1512.777	5781200	124.2413	1.0000	1.0000		
300	240	0.58	0.50	1.0009	0.5013	3.0121	0.0007	0.0048	0.0418	0.9524	0.9440	0.9536	2252.779	14120813	246.9324	1.0000	1.0000
		0.25	1.0009	3.0115	0.0007	0.0012	0.0419	0.9525	0.9437	0.9534	3428.872	56422299	242.0887	1.0000	1.0000		
		0.67	0.50	1.0005	0.5010	3.0121	0.0006	0.0074	0.0349	0.9510	0.9378	0.9552	2231.089	10664789	251.2180	1.0000	1.0000
		0.25	1.0003	3.0106	0.0006	0.0019	0.0349	0.9512	0.9375	0.9547	2885.223	42583191	248.4842	1.0000	1.0000		
$P = (1, 0, 0)$ (equivalently, $\alpha = (0, 0, 1)$)																	
$\beta = 1.00$																	
50	40	0.20	0.50	1.2515	0.5051	1.0918	1.9512	0.0120	0.0852	0.8411	0.9175	0.9587	316.229	87108	8.3163	0.0544	0.0413
		0.25	1.2528	1.0935	1.9698	1.0809	1.9698	0.0030	0.0891	0.8408	0.9177	0.9588	784.788	348758	5.8252	0.0555	0.0412
		0.30	0.50	1.1322	0.5078	1.0809	0.7399	0.0122	0.0721	0.8522	0.9303	0.9549	272.773	81434	9.1538	0.0547	0.0451
		0.25	1.1325	1.0811	0.7407	1.0811	0.7407	0.0031	0.0727	0.8524	0.9301	0.9546	635.509	325287	7.0756	0.0549	0.0454
100	80	0.20	0.50	1.0944	0.5023	1.0445	0.3883	0.0059	0.0328	0.8836	0.9325	0.9580	519.661	394497	16.6456	0.0486	0.0420
		0.25	1.0929	1.0445	0.3811	1.0445	0.3811	0.0015	0.0324	0.8836	0.9334	0.9584	1336.542	1577176	11.6548	0.0483	0.0416
		0.30	0.50	1.0585	0.5043	1.0378	0.2126	0.0059	0.0290	0.8880	0.9415	0.9553	472.992	425519	18.3108	0.0495	0.0447
		0.25	1.0584	1.0379	0.2123	1.0379	0.2123	0.0015	0.0290	0.8880	0.9413	0.9553	1129.226	1701662	14.1512	0.0495	0.0447
Individual components																	
150	120	0.07	0.50	1.1328	0.5034	1.0247	0.4561	0.0028	0.0154	0.9052	0.9489	0.9578	639.110	532273	65.3417	0.0507	0.0422
		0.25	1.1329	1.0248	0.4561	1.0248	0.4561	0.0007	0.0154	0.9051	0.9491	0.9580	2007.437	2127297	38.0455	0.0506	0.0420
		0.11	0.50	1.0661	0.5059	1.0294	0.1911	0.0034	0.0185	0.9086	0.9520	0.9584	618.226	602707	68.3863	0.0487	0.0416
		0.25	1.0661	1.0293	0.1913	1.0293	0.1913	0.0008	0.0185	0.9084	0.9522	0.9583	1900.393	2411475	42.5914	0.0487	0.0417
300	240	0.07	0.50	1.0504	0.5014	1.0128	0.1224	0.0013	0.0069	0.9279	0.9496	0.9561	1247.139	3591834	130.6117	0.0487	0.0439
		0.25	1.0512	1.0125	0.1228	1.0125	0.1228	0.0003	0.0069	0.9290	0.9487	0.9565	3920.454	14304570	76.1014	0.0477	0.0435
		0.11	0.50	1.0275	0.5023	1.0149	0.0768	0.0016	0.0083	0.9293	0.9508	0.9566	1209.028	4258445	136.6180	0.0489	0.0434
		0.25	1.0291	1.0143	0.0778	1.0143	0.0778	0.0004	0.0083	0.9305	0.9499	0.9559	3722.958	16937720	85.1886	0.0491	0.0441

(continued)

Table 35.3 (continued)

m	r	$F(\tau)$	θ_2	$E(\hat{\theta}_1)$	$E(\hat{\theta}_2)$	$E(\hat{\beta})$	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	CP_{θ_1}	CP_{θ_2}	CP_{β}	$tr(\mathbf{I})$	$det(\mathbf{I})$	TTT	$\frac{Error}{PowerL}$	$\frac{Error}{Powers}$	
Individual components																		
$\beta = 2.00$																		
50	40	0.20	0.50	1.1415	0.5194	2.2039	0.9091	0.0089	0.5062	0.8643	0.9562	0.9469	459.798	43632	18.1694	0.7901	0.4266	
			0.25	1.1396	0.2600	2.2099	0.9078	0.0023	0.5329	0.8629	0.9572	0.9465	1220.578	174505	15.4250	0.7908	0.4271	
	0.30	0.50	1.0538	0.5160	2.1554	0.1935	0.0107	0.3394	0.8834	0.9479	0.9522	371.659	55623	19.5244	0.8552	0.6531		
		0.25	1.0537	0.2580	2.1550	0.1937	0.0027	0.3355	0.8836	0.9487	0.9523	936.349	222171	17.4536	0.8553	0.6530		
100	80	0.20	0.50	1.0467	0.5100	2.1017	0.1330	0.0038	0.2002	0.8965	0.9589	0.9532	741.072	305730	36.3656	0.9711	0.8900	
			0.25	1.0468	0.2550	2.1010	0.1328	0.0010	0.1979	0.8965	0.9589	0.9530	2199.010	1222197	30.8628	0.9710	0.8899	
	0.30	0.50	1.0229	0.5081	2.0721	0.0567	0.0049	0.1362	0.9118	0.9479	0.9540	626.663	406354	39.0671	0.9896	0.9657		
		0.25	1.0234	0.2539	2.0715	0.0567	0.0012	0.1357	0.9125	0.9479	0.9540	1710.446	1622398	34.9170	0.9895	0.9656		
Individual components																		
150	120	0.07	0.50	1.0773	0.5124	2.0871	0.1772	0.0057	0.1718	0.9112	0.9435	0.9470	1309.312	452987	81.5400	0.9986	0.9905	
			0.25	1.0831	0.2560	2.0767	0.1822	0.0014	0.1684	0.9108	0.9422	0.9444	4636.109	1807240	60.7287	0.9986	0.9903	
	0.11	0.50	1.0398	0.5140	2.0880	0.0854	0.0065	0.1661	0.9114	0.9440	0.9492	1176.103	577210	86.4476	0.9982	0.9811		
		0.25	1.0384	0.2575	2.0862	0.0854	0.0017	0.1699	0.9080	0.9439	0.9482	4026.939	2315477	68.0755	0.9983	0.9810		
300	240	0.07	0.50	1.0302	0.5065	2.0443	0.0592	0.0024	0.0726	0.9324	0.9482	0.9504	2503.396	3530017	163.1042	1.0000	1.0000	
			0.25	1.0359	0.2529	2.0311	0.0604	0.0006	0.0703	0.9357	0.9470	0.9470	9106.935	14066663	121.4507	1.0000	1.0000	
	0.11	0.50	1.0182	0.5068	2.0421	0.0350	0.0029	0.0717	0.9335	0.9460	0.9498	2250.249	4504040	172.9271	1.0000	1.0000		
		0.25	1.0181	0.2538	2.0383	0.0360	0.0008	0.0733	0.9316	0.9440	0.9489	7913.862	18028763	136.1829	1.0000	1.0000		
$\beta = 3.00$																		
50	40	0.20	0.50	1.0934	0.5262	3.3087	0.5752	0.0152	1.2948	0.8762	0.9385	0.9470	741.026	40912	24.7592	0.9766	0.7738	
			0.25	1.0923	0.2635	3.3151	0.5775	0.0039	1.3489	0.8755	0.9397	0.9469	1735.444	163370	22.3366	0.9764	0.7738	
	0.30	0.50	1.0298	0.5196	3.2243	0.0818	0.0146	0.7895	0.8956	0.9393	0.9546	598.258	57471	26.0999	0.9925	0.9489		
		0.25	1.0295	0.2599	3.2236	0.0819	0.0036	0.7806	0.8953	0.9402	0.9551	1311.318	229641	24.3421	0.9924	0.9492		
100	80	0.20	0.50	1.0278	0.5134	3.1523	0.0610	0.0064	0.5003	0.9057	0.9446	0.9543	1163.339	318066	49.5464	0.9996	0.9914	
			0.25	1.0252	0.2573	3.1563	0.0601	0.0016	0.5013	0.9039	0.9468	0.9534	3059.422	1273989	44.6856	0.9996	0.9917	
	0.30	0.50	1.0127	0.5096	3.1028	0.0245	0.0066	0.3149	0.9193	0.9447	0.9534	996.697	450121	52.2150	1.0000	0.9998		
		0.25	1.0124	0.2549	3.1034	0.0244	0.0017	0.3157	0.9190	0.9453	0.9532	2350.645	1797871	48.6961	1.0000	0.9998		
Individual components																		
150	120	0.07	0.50	1.0606	0.5195	3.1676	0.1110	0.0117	0.6428	0.9072	0.9295	0.9387	2190.622	434468	95.0956	1.0000	0.9904	
			0.25	1.0597	0.2604	3.1597	0.1107	0.0031	0.6676	0.8988	0.9310	0.9364	7086.092	1758013	78.4721	1.0000	0.9924	
	0.11	0.50	1.0291	0.5184	3.1502	0.0484	0.0106	0.5185	0.9130	0.9382	0.9474	1914.904	608880	99.9498	1.0000	0.9974		
		0.25	1.0224	0.2608	3.1604	0.0463	0.0028	0.5369	0.9045	0.9414	0.9477	5916.904	2459854	85.7452	1.0000	0.9973		
300	240	0.07	0.50	1.0208	0.5108	3.0876	0.0362	0.0049	0.2642	0.9301	0.9440	0.9476	4001.245	3506123	190.1967	1.0000	0.9999	
			0.25	1.0257	0.2542	3.0575	0.0361	0.0012	0.2436	0.9336	0.9434	0.9462	13703.801	14104011	156.9594	1.0000	0.9998	
	0.11	0.50	1.0116	0.5094	3.0741	0.0188	0.0046	0.2174	0.9320	0.9448	0.9504	3542.330	4882751	199.9099	1.0000	1.0000		
		0.25	1.0128	0.2547	3.0578	0.0190	0.0012	0.2116	0.9320	0.9461	0.9500	11499.303	19596548	171.5234	1.0000	1.0000		

$P = (0, 2/3, 1/3)$ (equivalently, $\alpha = (1, 1, -1)$)

$\beta = 1.00$																	
50	40	0.20	0.50	1.0733	0.5434	1.1028	0.2273	0.0298	0.1101	0.8719	0.9486	0.9469	267.554	68787	34.3380	0.0577	0.0531
			0.25	1.0738	0.2720	1.1038	0.2296	0.0077	0.1152	0.8719	0.9484	0.9467	696.355	274434	27.0337	0.0581	0.0533
		0.30	0.50	1.0390	0.5364	1.0734	0.0886	0.0287	0.0687	0.9008	0.9467	0.9535	238.105	91923	37.1446	0.0494	0.0465
			0.25	1.0390	0.2683	1.0735	0.0887	0.0072	0.0688	0.9008	0.9468	0.9535	591.447	367100	31.2322	0.0494	0.0465
100	80	0.20	0.50	1.0346	0.5190	1.0465	0.0734	0.0117	0.0410	0.9069	0.9493	0.9510	483.234	501724	68.6505	0.0529	0.0490
			0.25	1.0351	0.2595	1.0460	0.0738	0.0029	0.0410	0.9074	0.9493	0.9504	1315.879	2000498	54.0552	0.0531	0.0496
		0.30	0.50	1.0184	0.5169	1.0348	0.0327	0.0125	0.0288	0.9246	0.9490	0.9522	440.811	677267	74.2498	0.0495	0.0478
			0.25	1.0177	0.2587	1.0350	0.0327	0.0031	0.0288	0.9242	0.9497	0.9525	1113.444	2706051	62.4423	0.0493	0.0475
Individual components																	
150	120	0.35	0.50	1.0158	0.5090	1.0222	0.0386	0.0067	0.0166	0.9277	0.9478	0.9528	506.308	1054785	86.0491	0.0485	0.0472
			0.25	1.0164	0.2545	1.0219	0.0388	0.0017	0.0167	0.9272	0.9479	0.9525	1343.364	4210569	69.1148	0.0487	0.0475
		0.44	0.50	1.0106	0.5087	1.0174	0.0241	0.0079	0.0133	0.9359	0.9456	0.9563	465.343	1235254	92.8288	0.0460	0.0437
			0.25	1.0106	0.2543	1.0174	0.0240	0.0020	0.0133	0.9360	0.9457	0.9565	1145.223	4935271	79.2727	0.0462	0.0435
300	240	0.35	0.50	1.0093	0.5020	1.0095	0.0183	0.0032	0.0078	0.9395	0.9475	0.9501	995.826	8046459	171.8679	0.0496	0.0499
			0.25	1.0103	0.2518	1.0092	0.0183	0.0008	0.0077	0.9413	0.9490	0.9515	2645.544	31867925	138.2411	0.0483	0.0485
		0.44	0.50	1.0049	0.5021	1.0085	0.0114	0.0038	0.0064	0.9411	0.9474	0.9506	913.622	9539016	185.4267	0.0480	0.0494
			0.25	1.0059	0.2517	1.0076	0.0112	0.0009	0.0063	0.9447	0.9506	0.9519	2243.748	37783222	158.5502	0.0471	0.0481
$\beta = 2.00$																	
50	40	0.20	0.50	1.0400	0.5456	2.2126	0.0762	0.0368	0.5640	0.8880	0.9386	0.9470	533.992	79241	39.1075	0.7237	0.4218
			0.25	1.0383	0.2736	2.2183	0.0738	0.0095	0.5911	0.8868	0.9400	0.9471	1295.361	315816	35.1244	0.7251	0.4219
		0.30	0.50	1.0172	0.5313	2.1379	0.0212	0.0275	0.2963	0.9179	0.9458	0.9544	445.776	115168	41.0958	0.8412	0.6728
			0.25	1.0168	0.2659	2.1392	0.0212	0.0069	0.2964	0.9175	0.9472	0.9548	1013.633	459358	38.0999	0.8429	0.6732
100	80	0.20	0.50	1.0173	0.5200	2.0953	0.0214	0.0145	0.2088	0.9180	0.9446	0.9488	889.973	628116	78.2240	0.9417	0.8366
			0.25	1.0155	0.2603	2.0974	0.0206	0.0036	0.2086	0.9179	0.9480	0.9501	2318.736	2509476	70.2462	0.9436	0.8395
		0.30	0.50	1.0078	0.5148	2.0659	0.0079	0.0122	0.1253	0.9351	0.9474	0.9548	789.255	902616	82.1870	0.9859	0.9648
			0.25	1.0072	0.2576	2.0671	0.0078	0.0030	0.1249	0.9351	0.9495	0.9558	1840.061	3600933	76.1981	0.9866	0.9655
Individual components																	
150	120	0.35	0.50	1.0073	0.5079	2.0393	0.0099	0.0067	0.0736	0.9390	0.9482	0.9512	790.438	1436667	104.4373	0.9998	0.9994
			0.25	1.0068	0.2541	2.0394	0.0099	0.0017	0.0740	0.9386	0.9482	0.9505	2188.843	5743864	95.0540	0.9998	0.9994
		0.44	0.50	1.0047	0.5065	2.0297	0.0059	0.0072	0.0553	0.9447	0.9467	0.9561	712.065	1671478	109.2360	1.0000	1.0000
			0.25	1.0044	0.2534	2.0302	0.0059	0.0018	0.0554	0.9441	0.9483	0.9561	1748.293	6668590	102.2666	1.0000	1.0000
300	240	0.35	0.50	1.0049	0.5026	2.0158	0.0048	0.0032	0.0345	0.9471	0.9452	0.9514	1524.016	11403138	208.8924	1.0000	1.0000
			0.25	1.0033	0.2522	2.0194	0.0049	0.0008	0.0355	0.9448	0.9457	0.9496	4255.841	45461716	190.1484	1.0000	1.0000
		0.44	0.50	1.0028	0.5019	2.0134	0.0028	0.0035	0.0264	0.9484	0.9475	0.9507	1376.072	13246441	218.4917	1.0000	1.0000
			0.25	1.0021	0.2520	2.0148	0.0028	0.0009	0.0267	0.9480	0.9498	0.9507	3375.124	52522791	204.5928	1.0000	1.0000

(continued)

Table 35.3 (continued)

<i>m</i>	<i>r</i>	<i>F</i> (τ)	θ_2	$E(\hat{\theta}_1)$	$E(\hat{\theta}_2)$	$E(\hat{\beta})$	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	CP_{θ_1}	CP_{θ_2}	CP_{β}	$tr(\mathbf{I})$	$det(\mathbf{I})$	TTT	$\frac{Error}{Power_L}$	$\frac{Error}{Powers}$	
Individual components																		
$\beta = 3.00$																		
50	40	0.20	0.50	1.0249	0.5460	3.3171	0.0368	0.0395	1.3288	0.8945	0.9426	0.9504	967.052	90792	41.8913	0.9527	0.7644	
			0.25	1.0213	0.2746	3.3363	0.0348	0.0101	1.4102	0.8928	0.9472	0.9521	1932.978	361653	39.1307	0.9548	0.7699	
100	80	0.30	0.50	1.0103	0.5295	3.2016	0.0087	0.0273	0.6749	0.9236	0.9464	0.9580	813.010	132468	43.3807	0.9899	0.9500	
			0.25	1.0091	0.2654	3.2074	0.0082	0.0069	0.6770	0.9229	0.9484	0.9584	1498.240	528256	41.3627	0.9908	0.9520	
300	240	0.30	0.50	1.0104	0.5200	3.1418	0.0093	0.0156	0.4936	0.9213	0.9475	0.9518	1568.584	729504	83.7921	0.9983	0.9858	
			0.25	1.0073	0.2615	3.1567	0.0087	0.0039	0.4941	0.9190	0.9542	0.9555	3328.877	2904815	78.2671	0.9984	0.9868	
150	120	0.30	0.50	1.0048	0.5137	3.0951	0.0034	0.0122	0.2853	0.9378	0.9488	0.9561	1428.975	1044000	86.7636	1.0000	0.9999	
			0.25	1.0041	0.2575	3.0996	0.0033	0.0030	0.2852	0.9370	0.9522	0.9575	2675.779	4155917	82.7268	1.0000	0.9999	
Individual components																		
$P = (1/3, 2/3, 0)$ (equivalently, $\mathbf{a} = (0, 2, -1)$)																		
$\beta = 1.00$																		
50	40	0.20	0.50	1.1631	0.5180	1.0984	0.7661	0.0106	0.0993	0.8605	0.9548	0.9607	264.639	52175	18.3102	0.0537	0.0393	
			0.25	1.1627	0.2592	1.0996	0.7668	0.0027	0.1016	0.8606	0.9551	0.9605	747.042	208219	13.4828	0.0542	0.0395	
100	80	0.30	0.50	1.0752	0.5205	1.0858	0.2840	0.0139	0.0819	0.8740	0.9525	0.9561	230.296	60804	20.1751	0.0541	0.0439	
			0.25	1.0747	0.2603	1.0861	0.2835	0.0035	0.0824	0.8738	0.9526	0.9563	613.195	242980	16.2655	0.0543	0.0437	
300	240	0.30	0.50	1.0716	0.5085	1.0456	0.2082	0.0047	0.0381	0.8980	0.9526	0.9588	466.503	327938	36.6267	0.0512	0.0412	
			0.25	1.0723	0.2543	1.0455	0.2087	0.0012	0.0381	0.8988	0.9529	0.9590	1383.860	1307734	26.9654	0.0510	0.0410	
150	120	0.30	0.50	1.0351	0.5101	1.0402	0.0989	0.0062	0.0328	0.9051	0.9550	0.9553	415.636	411636	40.3309	0.0500	0.0447	
			0.25	1.0355	0.2551	1.0399	0.0992	0.0016	0.0328	0.9054	0.9547	0.9552	1146.719	1643221	32.5151	0.0498	0.0448	
Individual components																		
50	40	0.17	0.50	1.0370	0.5081	1.0304	0.1099	0.0044	0.0200	0.9124	0.9514	0.9576	586.741	703155	73.0819	0.0485	0.0424	
			0.25	1.0370	0.2540	1.0303	0.1100	0.0011	0.0200	0.9124	0.9511	0.9576	1741.120	2811555	49.6393	0.0483	0.0424	
100	80	0.25	0.50	1.0223	0.5094	1.0281	0.0664	0.0054	0.0192	0.9162	0.9527	0.9567	549.213	851835	78.8981	0.0486	0.0433	
			0.25	1.0225	0.2547	1.0278	0.0664	0.0013	0.0192	0.9164	0.9525	0.9568	1555.347	3403144	58.3673	0.0486	0.0432	
300	240	0.25	0.50	1.0172	0.5027	1.0150	0.0499	0.0021	0.0091	0.9324	0.9483	0.9531	1151.634	5148040	145.9052	0.0501	0.0469	
			0.25	1.0177	0.2520	1.0146	0.0494	0.0005	0.0091	0.9330	0.9490	0.9536	3423.549	20445269	99.2845	0.0495	0.0464	
150	120	0.25	0.50	1.0085	0.5034	1.0146	0.0300	0.0026	0.0089	0.9330	0.9497	0.9550	1079.977	6398872	157.5040	0.0495	0.0450	
			0.25	1.0106	0.2523	1.0134	0.0298	0.0006	0.0088	0.9351	0.9521	0.9547	3063.772	25306209	116.7527	0.0493	0.0453	

Individual components																	
$\beta = 2.00$																	
50	40	0.20	0.50	1.0898	0.5307	2.2155	0.2826	0.0199	0.5600	0.8770	0.9322	0.9501	505.816	45199	27.6572	0.7379	0.3553
		0.25	1.0898	0.2658	2.2193	0.2841	0.0051	0.5876	0.8763	0.9332	0.9499	1386.875	180201	24.0631	0.7374	0.3542	
		0.30	1.0322	0.5252	2.1653	0.0715	0.0191	0.3754	0.8951	0.9423	0.9541	415.399	64309	29.5655	0.8154	0.5914	
		0.25	1.0321	0.2627	2.1655	0.0716	0.0048	0.3763	0.8945	0.9424	0.9538	1058.508	256778	26.9139	0.8152	0.5905	
100	80	0.20	0.50	1.0368	0.5143	2.0986	0.0686	0.0084	0.2177	0.9099	0.9419	0.9524	840.684	349456	55.3308	0.9530	0.8295
		0.25	1.0358	0.2574	2.0996	0.0684	0.0021	0.2188	0.9087	0.9434	0.9525	2526.776	1396752	48.1251	0.9528	0.8298	
		0.30	1.0152	0.5121	2.0764	0.0255	0.0085	0.1509	0.9193	0.9465	0.9516	713.434	503834	59.1269	0.9812	0.9462	
		0.25	1.0148	0.2562	2.0767	0.0253	0.0021	0.1512	0.9193	0.9482	0.9514	1929.163	2012084	53.8191	0.9814	0.9468	
Individual components																	
150	120	0.17	0.50	1.0213	0.5126	2.0721	0.0409	0.0067	0.1325	0.9197	0.9476	0.9517	1028.639	792781	92.4611	0.9963	0.9830
		0.25	1.0190	0.2570	2.0730	0.0408	0.0017	0.1339	0.9176	0.9485	0.9513	3358.733	3173706	77.1011	0.9961	0.9835	
		0.25	1.0114	0.5104	2.0551	0.0196	0.0065	0.0992	0.9293	0.9511	0.9521	900.794	1090561	98.4152	0.9981	0.9936	
		0.25	1.0103	0.2556	2.0570	0.0195	0.0016	0.1003	0.9289	0.9519	0.9521	2759.619	4353105	86.0448	0.9982	0.9937	
300	240	0.17	0.50	1.0107	0.5060	2.0334	0.0183	0.0031	0.0595	0.9347	0.9471	0.9490	1974.298	6237878	184.9457	1.0000	1.0000
		0.25	1.0086	0.2538	2.0363	0.0189	0.0008	0.0627	0.9303	0.9442	0.9466	6605.545	24971481	154.2072	1.0000	1.0000	
		0.25	1.0059	0.5048	2.0255	0.0091	0.0031	0.0459	0.9397	0.9494	0.9539	1733.247	8630490	196.8620	1.0000	0.9999	
		0.25	1.0025	0.2533	2.0326	0.0094	0.0008	0.0486	0.9351	0.9473	0.9511	5393.091	34619596	172.0494	1.0000	0.9999	
$\beta = 3.00$																	
50	40	0.20	0.50	1.0585	0.5349	3.3230	0.1505	0.0265	1.3693	0.8864	0.9309	0.9517	875.430	49012	32.9633	0.9636	0.7206
		0.25	1.0516	0.2691	3.3432	0.1281	0.0069	1.4711	0.8832	0.9339	0.9515	2041.822	196054	30.1841	0.9639	0.7255	
		0.30	1.0186	0.5266	3.2407	0.0299	0.0225	0.8649	0.9042	0.9421	0.9541	724.384	72831	34.5819	0.9891	0.9292	
		0.25	1.0175	0.2636	3.2435	0.0293	0.0056	0.8561	0.9038	0.9441	0.9551	1529.847	290669	32.6107	0.9896	0.9306	
100	80	0.20	0.50	1.0225	0.5159	3.1459	0.0312	0.0111	0.5250	0.9145	0.9425	0.9547	1399.995	396433	65.9417	0.9990	0.9864
		0.25	1.0158	0.2600	3.1663	0.0290	0.0028	0.5427	0.9101	0.9484	0.9555	3563.701	1586355	60.3783	0.9992	0.9872	
		0.30	1.0088	0.5127	3.1110	0.0108	0.0100	0.3501	0.9258	0.9455	0.9511	1224.066	583399	69.1596	0.9999	0.9995	
		0.25	1.0070	0.2570	3.1158	0.0104	0.0025	0.3497	0.9252	0.9487	0.9522	2711.279	2328892	65.2063	0.9999	0.9995	
Individual components																	
150	120	0.17	0.50	1.0137	0.5141	3.1107	0.0196	0.0086	0.3480	0.9227	0.9451	0.9508	1645.521	908634	105.4284	1.0000	0.9996
		0.25	1.0081	0.2586	3.1218	0.0180	0.0022	0.3528	0.9183	0.9503	0.9532	4766.129	3650526	93.9785	1.0000	0.9996	
		0.25	1.0071	0.5105	3.0806	0.0087	0.0073	0.2384	0.9354	0.9510	0.9539	1451.037	1290868	110.4844	1.0000	0.9999	
		0.25	1.0053	0.2560	3.0860	0.0085	0.0018	0.2401	0.9343	0.9523	0.9544	3831.111	5151693	101.5799	1.0000	0.9999	
300	240	0.17	0.50	1.0069	0.5066	3.0509	0.0087	0.0041	0.1549	0.9380	0.9452	0.9513	3094.113	7280370	210.8834	1.0000	1.0000
		0.25	1.0046	0.2544	3.0515	0.0088	0.0011	0.1573	0.9336	0.9440	0.9505	9249.482	29125391	187.9858	1.0000	1.0000	
		0.25	1.0038	0.5049	3.0369	0.0041	0.0035	0.1101	0.9418	0.9495	0.9539	2756.595	10334103	220.9933	1.0000	1.0000	
		0.25	1.0013	0.2537	3.0465	0.0042	0.0009	0.1150	0.9377	0.9494	0.9526	7398.173	41214800	203.1651	1.0000	1.0000	

Table 35.4 Relative efficiency of the MSEs, TTT, trace, and determinant of observed Fisher information matrix for some three-component coherent systems and component life tests. The quantity $F(\tau)$ in the third column corresponds to $F_T(\tau)$ in case of system lifetimes and corresponds to $F_X(\tau)$ in case of component lifetimes

m	r	$F(\tau)$	θ_2	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	$tr(\mathbf{I})$	$det(\mathbf{I})$	TTT
$\beta = 1.00$									
$P = (0, 0, 1)$ (equivalently, $a = (3, -3, 1)$)									
50	40	0.20	0.50	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		0.30	0.50	0.4553	0.7303	0.5913	0.8986	1.4357	1.0701
100	80	0.20	0.50	0.3502	0.3873	0.3874	1.8147	7.4033	2.0008
		0.30	0.50	0.1848	0.3264	0.2566	1.6562	10.4950	2.1405
Individual components									
150	120	0.58	0.50	0.2607	0.1618	0.0824	1.0727	10.2729	1.6962
		0.67	0.50	0.2101	0.2586	0.0697	0.9735	8.8785	1.7992
300	240	0.58	0.50	0.1284	0.0761	0.0391	2.0863	78.3679	3.3909
		0.67	0.50	0.1012	0.1176	0.0331	1.8829	65.8178	3.5957
$P = (1, 0, 0)$ (equivalently, $a = (0, 0, 1)$)									
50	40	0.20	0.50	37.9611	0.1660	0.7239	0.8443	0.6700	0.1359
		0.30	0.50	14.3949	0.1687	0.6126	0.7282	0.6263	0.1496
100	80	0.20	0.50	7.5545	0.0816	0.2787	1.3874	3.0341	0.2721
		0.30	0.50	4.1362	0.0816	0.2464	1.2628	3.2727	0.2993
Individual components									
150	120	0.07	0.50	8.8735	0.0387	0.1308	1.7063	4.0937	1.0682
		0.11	0.50	3.7179	0.0470	0.1572	1.6505	4.6355	1.1179
300	240	0.07	0.50	2.3813	0.0180	0.0586	3.3296	27.6250	2.1352
		0.11	0.50	1.4942	0.0221	0.0705	3.2278	32.7520	2.2333
$P = (0, 2/3, 1/3)$ (equivalently, $a = (1, 1, -1)$)									
50	40	0.20	0.50	4.4222	0.4122	0.9354	0.7143	0.5290	0.5613
		0.30	0.50	1.7237	0.3970	0.5837	0.6357	0.7070	0.6072
100	80	0.20	0.50	1.4280	0.1618	0.3483	1.2901	3.8588	1.1223
		0.30	0.50	0.6362	0.1729	0.2447	1.1769	5.2089	1.2138
Individual components									
150	120	0.35	0.50	0.7510	0.0927	0.1410	1.3517	8.1124	1.4067
		0.44	0.50	0.4689	0.1093	0.1130	1.2424	9.5004	1.5175
300	240	0.35	0.50	0.3560	0.0443	0.0663	2.6586	61.8858	2.8096
		0.44	0.50	0.2218	0.0526	0.0544	2.4392	73.3652	3.0312
$P = (1/3, 2/3, 0)$ (equivalently, $a = (0, 2, -1)$)									
50	40	0.20	0.50	14.9047	0.1466	0.8437	0.7065	0.4013	0.2993
		0.30	0.50	5.5253	0.1923	0.6958	0.6148	0.4676	0.3298
100	80	0.20	0.50	4.0506	0.0650	0.3237	1.2455	2.5222	0.5988
		0.30	0.50	1.9241	0.0858	0.2787	1.1097	3.1659	0.6593
Individual components									
150	120	0.17	0.50	2.1381	0.0609	0.1699	1.5665	5.4080	1.1947
		0.25	0.50	1.2918	0.0747	0.1631	1.4663	6.5515	1.2898
300	240	0.17	0.50	0.9708	0.0290	0.0773	3.0746	39.5939	2.3852
		0.25	0.50	0.5837	0.0360	0.0756	2.8833	49.2141	2.5748
$P = (0, 0, 1)$ (equivalently, $a = (3, -3, 1)$)									
50	40	0.20	0.25	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		0.30	0.25	0.4563	0.7293	0.5843	0.8348	1.4316	1.1255
100	80	0.20	0.25	0.3472	0.3867	0.3820	1.8247	7.4248	2.0002
		0.30	0.25	0.1865	0.3260	0.2521	1.5230	10.5041	2.2510
Individual components									
150	120	0.58	0.25	0.2679	0.1602	0.0818	0.8720	10.3194	1.8720
		0.67	0.25	0.2163	0.2597	0.0691	0.6784	8.8965	2.0582

(continued)

Table 35.4 (continued)

m	r	$F(\tau)$	θ_2	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	$tr(\mathbf{I})$	$det(\mathbf{I})$	TTT
Individual components									
300	240	0.58	0.25	0.1270	0.0773	0.0396	1.6816	78.9226	3.7430
		0.67	0.25	0.1032	0.1160	0.0329	1.2823	66.2139	4.1154
$P = (1, 0, 0)$ (equivalently, $\alpha = (0, 0, 1)$)									
50	40	0.20	0.25	39.0833	0.1657	0.7513	0.8237	0.6761	0.1139
		0.30	0.25	14.6964	0.1713	0.6130	0.6670	0.6306	0.1383
100	80	0.20	0.25	7.5615	0.0829	0.2732	1.4028	3.0575	0.2279
		0.30	0.25	4.2123	0.0829	0.2445	1.1852	3.2988	0.2767
Individual components									
150	120	0.07	0.25	9.0496	0.0387	0.1298	2.1069	4.1239	0.7438
		0.11	0.25	3.7956	0.0442	0.1560	1.9946	4.6749	0.8327
300	240	0.07	0.25	2.4365	0.0166	0.0582	4.1148	27.7306	1.4878
		0.11	0.25	1.5437	0.0221	0.0700	3.9075	32.8352	1.6655
$P = (0, 2/3, 1/3)$ (equivalently, $\alpha = (1, 1, -1)$)									
50	40	0.20	0.25	4.5556	0.4254	0.9713	0.7309	0.5320	0.5285
		0.30	0.25	1.7599	0.3978	0.5801	0.6208	0.7117	0.6106
100	80	0.20	0.25	1.4643	0.1602	0.3457	1.3811	3.8781	1.0568
		0.30	0.25	0.6488	0.1713	0.2428	1.1686	5.2459	1.2208
Individual components									
150	120	0.35	0.25	0.7698	0.0939	0.1408	1.4099	8.1625	1.3512
		0.44	0.25	0.4762	0.1105	0.1121	1.2020	9.5674	1.5498
300	240	0.35	0.25	0.3631	0.0442	0.0649	2.7767	61.7787	2.7027
		0.44	0.25	0.2222	0.0497	0.0531	2.3550	73.2460	3.0997
$P = (1/3, 2/3, 0)$ (equivalently, $\alpha = (0, 2, -1)$)									
50	40	0.20	0.25	15.2143	0.1492	0.8567	0.7841	0.4037	0.2636
		0.30	0.25	5.6250	0.1934	0.6948	0.6436	0.4710	0.3180
100	80	0.20	0.25	4.1409	0.0663	0.3212	1.4525	2.5352	0.5272
		0.30	0.25	1.9683	0.0884	0.2766	1.2036	3.1855	0.6357
Individual components									
150	120	0.17	0.25	2.1825	0.0608	0.1686	1.8274	5.4504	0.9705
		0.25	0.25	1.3175	0.0718	0.1619	1.6324	6.5973	1.1411
300	240	0.17	0.25	0.9802	0.0276	0.0767	3.5932	39.6349	1.9411
		0.25	0.25	0.5913	0.0331	0.0742	3.2156	49.0583	2.2826
$\beta = 2.00$									
$P = (0, 0, 1)$ (equivalently, $\alpha = (3, -3, 1)$)									
50	40	0.20	0.50	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		0.30	0.50	0.3813	0.6525	0.5147	0.8490	1.4368	1.0403
100	80	0.20	0.50	0.3094	0.4034	0.3812	1.6758	7.6775	2.0006
		0.30	0.50	0.1583	0.3000	0.2236	1.5158	10.9245	2.0808
Individual components									
150	120	0.58	0.50	0.2374	0.1746	0.0686	0.8336	10.2178	2.1675
		0.67	0.50	0.1942	0.2932	0.0573	0.7867	8.2826	2.2275
300	240	0.58	0.50	0.1151	0.0847	0.0331	1.6110	78.8412	4.3355
		0.67	0.50	0.0935	0.1288	0.0271	1.5200	61.4296	4.4556
$P = (1, 0, 0)$ (equivalently, $\alpha = (0, 0, 1)$)									
50	40	0.20	0.50	65.4029	0.1508	0.8904	0.6141	0.2673	0.3403
		0.30	0.50	13.9209	0.1814	0.5970	0.4964	0.3408	0.3656
100	80	0.20	0.50	9.5683	0.0644	0.3522	0.9898	1.8730	0.6810
		0.30	0.50	4.0791	0.0831	0.2396	0.8370	2.4895	0.7316
Individual components									
150	120	0.07	0.50	12.7482	0.0966	0.3022	1.7488	2.7752	1.5271
		0.11	0.50	6.1439	0.1102	0.2922	1.5708	3.5362	1.6190
300	240	0.07	0.50	4.2590	0.0407	0.1277	3.3436	21.6263	3.0546
		0.11	0.50	2.5180	0.0492	0.1261	3.0055	27.5936	3.2385

(continued)

Table 35.4 (continued)

m	r	$F(\tau)$	θ_2	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	$tr(\mathbf{I})$	$det(\mathbf{I})$	TTT
$P = (0, 2/3, 1/3)$ (equivalently, $\mathbf{a} = (1, 1, -1)$)									
50	40	0.20	0.50	5.4820	0.6237	0.9921	0.7132	0.4855	0.7324
		0.30	0.50	1.5252	0.4661	0.5212	0.5954	0.7056	0.7696
100	80	0.20	0.50	1.5396	0.2458	0.3673	1.1887	3.8481	1.4650
		0.30	0.50	0.5683	0.2068	0.2204	1.0542	5.5298	1.5392
Individual components									
150	120	0.35	0.50	0.7122	0.1136	0.1295	1.0557	8.8016	1.9559
		0.44	0.50	0.4245	0.1220	0.0973	0.9511	10.2401	2.0457
300	240	0.35	0.50	0.3453	0.0542	0.0607	2.0355	69.8602	3.9121
		0.44	0.50	0.2014	0.0593	0.0464	1.8379	81.1530	4.0919
$P = (1/3, 2/3, 0)$ (equivalently, $\mathbf{a} = (0, 2, -1)$)									
50	40	0.20	0.50	20.3309	0.3373	0.9850	0.6756	0.2769	0.5180
		0.30	0.50	5.1439	0.3237	0.6603	0.5548	0.3940	0.5537
100	80	0.20	0.50	4.9353	0.1424	0.3829	1.1228	2.1409	1.0362
		0.30	0.50	1.8345	0.1441	0.2654	0.9529	3.0867	1.1073
Individual components									
150	120	0.17	0.50	2.9424	0.1136	0.2331	1.3739	4.8569	1.7316
		0.25	0.50	1.4101	0.1102	0.1745	1.2031	6.6812	1.8431
300	240	0.17	0.50	1.3165	0.0525	0.1047	2.6369	38.2157	3.4636
		0.25	0.50	0.6547	0.0525	0.0807	2.3150	52.8738	3.6868
$P = (0, 0, 1)$ (equivalently, $\mathbf{a} = (3, -3, 1)$)									
50	40	0.20	0.25	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		0.30	0.25	0.4262	0.6621	0.5213	0.7803	1.4448	1.0656
100	80	0.20	0.25	0.3279	0.4069	0.3895	1.7260	7.6675	2.0004
		0.30	0.25	0.1803	0.2966	0.2244	1.3806	10.9519	2.1312
Individual components									
150	120	0.58	0.25	0.2705	0.1793	0.0698	0.7153	10.3851	2.2750
		0.67	0.25	0.2213	0.2966	0.0586	0.5593	8.4136	2.3724
300	240	0.58	0.25	0.1311	0.0828	0.0335	1.3640	80.0918	4.5512
		0.67	0.25	0.1066	0.1310	0.0283	1.0551	62.6725	4.7467
$P = (1, 0, 0)$ (equivalently, $\mathbf{a} = (0, 0, 1)$)									
50	40	0.20	0.25	74.4098	0.1586	0.9543	0.7320	0.2719	0.3132
		0.30	0.25	15.8770	0.1862	0.6008	0.5616	0.3461	0.3544
100	80	0.20	0.25	10.8852	0.0690	0.3544	1.3188	1.9041	0.6268
		0.30	0.25	4.6475	0.0828	0.2430	1.0258	2.5276	0.7091
Individual components									
150	120	0.07	0.25	14.9344	0.0966	0.3016	2.7804	2.8156	1.2333
		0.11	0.25	7.0000	0.1172	0.3043	2.4151	3.6074	1.3825
300	240	0.07	0.25	4.9508	0.0414	0.1259	5.4617	21.9152	2.4664
		0.11	0.25	2.9508	0.0552	0.1313	4.7462	28.0879	2.7656
$P = (0, 2/3, 1/3)$ (equivalently, $\mathbf{a} = (1, 1, -1)$)									
50	40	0.20	0.25	6.0492	0.6552	1.0586	0.7769	0.4920	0.7133
		0.30	0.25	1.7377	0.4759	0.5308	0.6079	0.7157	0.7737
100	80	0.20	0.25	1.6885	0.2483	0.3736	1.3906	3.9096	1.4265
		0.30	0.25	0.6393	0.2069	0.2237	1.1035	5.6101	1.5474
Individual components									
150	120	0.35	0.25	0.8115	0.1172	0.1325	1.3127	8.9487	1.9303
		0.44	0.25	0.4836	0.1241	0.0992	1.0485	10.3893	2.0768
300	240	0.35	0.25	0.4016	0.0552	0.0636	2.5524	70.8271	3.8615
		0.44	0.25	0.2295	0.0621	0.0478	2.0242	81.8279	4.1548
$P = (1/3, 2/3, 0)$ (equivalently, $\mathbf{a} = (0, 2, -1)$)									
50	40	0.20	0.25	23.2869	0.3517	1.0523	0.8318	0.2807	0.4887
		0.30	0.25	5.8689	0.3310	0.6739	0.6348	0.4000	0.5466

(continued)

Table 35.4 (continued)

m	r	$F(\tau)$	θ_2	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	$tr(\mathbf{I})$	$det(\mathbf{I})$	TTT
$P = (1/3, 2/3, 0)$ (equivalently, $\mathbf{a} = (0, 2, -1)$)									
100	80	0.20	0.25	5.6066	0.1448	0.3918	1.5154	2.1761	0.9773
		0.30	0.25	2.0738	0.1448	0.2708	1.1570	3.1347	1.0929
Individual components									
150	120	0.17	0.25	3.3443	0.1172	0.2398	2.0143	4.9445	1.5657
		0.25	0.25	1.5984	0.1103	0.1796	1.6550	6.7819	1.7474
300	240	0.17	0.25	1.5492	0.0552	0.1123	3.9616	38.9043	3.1316
		0.25	0.25	0.7705	0.0552	0.0870	3.2344	53.9356	3.4939
$\beta = 3.00$									
$P = (0, 0, 1)$ (equivalently, $\mathbf{a} = (3, -3, 1)$)									
50	40	0.20	0.50	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		0.30	0.50	0.3966	0.6492	0.5010	0.8632	1.4137	1.0278
100	80	0.20	0.50	0.3103	0.4071	0.3768	1.6518	7.7256	2.0004
		0.30	0.50	0.1724	0.2983	0.2172	1.5432	10.7998	2.0557
Individual components									
150	120	0.58	0.50	0.2586	0.1876	0.0663	0.8234	9.8810	2.3818
		0.67	0.50	0.2069	0.3208	0.0555	0.8125	7.8007	2.4231
300	240	0.58	0.50	0.1207	0.0901	0.0315	1.6003	75.9735	4.7644
		0.67	0.50	0.1034	0.1388	0.0263	1.5849	57.3792	4.8471
$P = (1, 0, 0)$ (equivalently, $\mathbf{a} = (0, 0, 1)$)									
50	40	0.20	0.50	99.1724	0.2852	0.9769	0.5264	0.2201	0.4777
		0.30	0.50	14.1034	0.2739	0.5957	0.4250	0.3092	0.5036
100	80	0.20	0.50	10.5172	0.1201	0.3775	0.8264	1.7113	0.9560
		0.30	0.50	4.2241	0.1238	0.2376	0.7080	2.4218	1.0074
Individual components									
150	120	0.07	0.50	19.1379	0.2195	0.4850	1.5561	2.3375	1.8348
		0.11	0.50	8.3448	0.1989	0.3912	1.3603	3.2759	1.9285
300	240	0.07	0.50	6.2414	0.0919	0.1993	2.8424	18.8638	3.6697
		0.11	0.50	3.2414	0.0863	0.1640	2.5164	26.2704	3.8571
$P = (0, 2/3, 1/3)$ (equivalently, $\mathbf{a} = (1, 1, -1)$)									
50	40	0.20	0.50	6.3448	0.7411	1.0026	0.6870	0.4885	0.8083
		0.30	0.50	1.5000	0.5122	0.5092	0.5775	0.7127	0.8370
100	80	0.20	0.50	1.6034	0.2927	0.3724	1.1143	3.9249	1.6167
		0.30	0.50	0.5862	0.2289	0.2153	1.0151	5.6170	1.6740
Individual components									
150	120	0.35	0.50	0.7414	0.1313	0.1277	0.9306	9.1158	2.2238
		0.44	0.50	0.4483	0.1351	0.0945	0.8724	10.3629	2.2932
300	240	0.35	0.50	0.3621	0.0638	0.0598	1.7821	72.7254	4.4481
		0.44	0.50	0.2069	0.0657	0.0449	1.6813	82.0973	4.5870
$P = (1/3, 2/3, 0)$ (equivalently, $\mathbf{a} = (0, 2, -1)$)									
50	40	0.20	0.50	25.9483	0.4972	1.0331	0.6219	0.2637	0.6360
		0.30	0.50	5.1552	0.4221	0.6526	0.5146	0.3918	0.6672
100	80	0.20	0.50	5.3793	0.2083	0.3961	0.9945	2.1329	1.2723
		0.30	0.50	1.8621	0.1876	0.2641	0.8695	3.1388	1.3344
Individual components									
150	120	0.17	0.50	3.3793	0.1614	0.2626	1.1689	4.8887	2.0342
		0.25	0.50	1.5000	0.1370	0.1799	1.0308	6.9452	2.1317
300	240	0.17	0.50	1.5000	0.0769	0.1169	2.1980	39.1702	4.0688
		0.25	0.50	0.7069	0.0657	0.0831	1.9582	55.6000	4.2639
$P = (0, 0, 1)$ (equivalently, $\mathbf{a} = (3, -3, 1)$)									
50	40	0.20	0.25	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		0.30	0.25	0.4151	0.6296	0.4821	0.7820	1.4222	1.0439
100	80	0.20	0.25	0.3019	0.4074	0.3713	1.6741	7.6638	2.0003
		0.30	0.25	0.1698	0.2889	0.2108	1.3853	10.7988	2.0879

(continued)

Table 35.4 (continued)

m	r	$F(\tau)$	θ_2	$MSE(\hat{\theta}_1)$	$MSE(\hat{\theta}_2)$	$MSE(\hat{\beta})$	$tr(\mathbf{I})$	$det(\mathbf{I})$	TTT
Individual components									
150	120	0.58	0.25	0.2830	0.1852	0.0646	0.7059	10.0502	2.4593
		0.67	0.25	0.2264	0.3185	0.0538	0.5962	7.9107	2.5246
300	240	0.58	0.25	0.1321	0.0889	0.0308	1.3514	77.2053	4.9192
		0.67	0.25	0.1132	0.1407	0.0256	1.1372	58.2686	5.0492
$P = (1, 0, 0)$ (equivalently, $\alpha = (0, 0, 1)$)									
50	40	0.20	0.25	108.9623	0.2889	0.9906	0.6840	0.2235	0.4539
		0.30	0.25	15.4528	0.2667	0.5733	0.5168	0.3142	0.4946
100	80	0.20	0.25	11.3396	0.1185	0.3681	1.2058	1.7433	0.9080
		0.30	0.25	4.6038	0.1259	0.2318	0.9265	2.4601	0.9895
Individual components									
150	120	0.07	0.25	20.8868	0.2296	0.4903	2.7929	2.4056	1.5945
		0.11	0.25	8.7358	0.2074	0.3943	2.3321	3.3659	1.7423
300	240	0.07	0.25	6.8113	0.0889	0.1789	5.4012	19.2992	3.1894
		0.11	0.25	3.5849	0.0889	0.1554	4.5323	26.8149	3.4853
$P = (0, 2/3, 1/3)$ (equivalently, $\alpha = (1, 1, -1)$)									
50	40	0.20	0.25	6.5660	0.7481	1.0356	0.7619	0.4949	0.7951
		0.30	0.25	1.5472	0.5111	0.4972	0.5905	0.7228	0.8405
100	80	0.20	0.25	1.6415	0.2889	0.3629	1.3120	3.9748	1.5904
		0.30	0.25	0.6226	0.2222	0.2094	1.0546	5.6867	1.6810
Individual components									
150	120	0.35	0.25	0.8113	0.1259	0.1246	1.1899	9.2383	2.2098
		0.44	0.25	0.4906	0.1333	0.0922	0.9649	10.5229	2.3193
300	240	0.35	0.25	0.3962	0.0667	0.0599	2.3004	73.6726	4.4197
		0.44	0.25	0.2453	0.0667	0.0449	1.8554	82.7799	4.6403
$P = (1/3, 2/3, 0)$ (equivalently, $\alpha = (0, 2, -1)$)									
50	40	0.20	0.25	24.1698	0.5111	1.0803	0.8048	0.2683	0.6133
		0.30	0.25	5.5283	0.4148	0.6287	0.6030	0.3977	0.6626
100	80	0.20	0.25	5.4717	0.2074	0.3985	1.4046	2.1707	1.2269
		0.30	0.25	1.9623	0.1852	0.2568	1.0686	3.1867	1.3250
Individual components									
150	120	0.17	0.25	3.3962	0.1630	0.2591	1.8785	4.9952	1.9096
		0.25	0.25	1.6038	0.1333	0.1763	1.5100	7.0493	2.0641
300	240	0.17	0.25	1.6604	0.0815	0.1155	3.6456	39.8536	3.8198
		0.25	0.25	0.7925	0.0667	0.0845	2.9159	56.3961	4.1283

efficiency. It is also of interest to note that the test of parallel systems in some cases results in more information (in trace) than the test of individual components even though the latter has a larger TTT.

Note that the exponential distribution is a special case of the Weibull distribution when the value of the shape parameter $\beta = 1$. Thus, it is possible to carry out a test of a hypothesis whether the data have come from an exponential distribution. In our simulation study, we calculate the Type-I error for testing the hypothesis $H_0 : \beta = 1$ based on log-likelihood ratio test and score test. For other values of β , we recorded the powers of the log-likelihood ratio test and score test, and these results are presented in Table 35.3. By comparing the likelihood ratio and score tests, we find that

the likelihood ratio test performs better in both Type-I error and power for most of the cases and hence is the one to be preferred!

Finally, we report the relative efficiency of MSEs, $tr(\mathbf{I})$, and $det(\mathbf{I})$ in Table 35.4 under the same θ_1 , θ_2 , and β , but different values of m , r , $F_T(\tau)$, and signature vectors. Here, the relative efficiency of MSE is defined as

$$RE = \frac{\text{MSE of the Model of Interest}}{\text{MSE of the Reference Model}}$$

where the models with $m = 50$, $r = 40$, $F_T(\tau) = 0.2$, and $p = (0, 0, 1)$ are the reference models. The relative efficiency of other quantities is defined similarly.

Table 35.5 MLEs of the parameters based on data in Table 35.1, their SEs, and 95% CIs based on the asymptotic approach using observed Fisher information matrix and the bootstrap method

Parm.	MLEs	Observed Fisher		Bootstrap	
		SE	Asymptotic CI	SE	Bootstrap CI
θ_1	14.6054	2.0939	(10.5012, 18.7095)	7.0973	(11.6166, 25.3188)
θ_2	4.8595	1.5253	(1.8700, 7.8490)	2.2765	(2.4757, 10.8768)
β	3.2160	0.9893	(1.2771, 5.1550)	1.7563	(1.6681, 7.8655)

Table 35.6 Log-likelihood ratio test and AIC for $\beta = 1$

$\ln L_0$	$\ln L_1$	$2(\ln L_1 - \ln L_0)$	$\chi_{0.05,1}$	AIC_0	AIC_1
-76.7407	-71.2022	11.0770	3.8415	157.4814	148.4044

35.6 Illustrative Example

For the data presented in Table 35.1, we consider a Type-II censored system lifetime data in hours (with $m=50$ and $r=30$) with signature vector $\mathbf{P} = (0, 1/2, 1/4, 1/4)$. In this case, we have $n_1=8$ and $n_2 = r - n_1=30 - 8 = 22$. From Equations (35.20) and (35.22), we first determine the initial estimates of θ_1 , θ_2 , and β to be $\tilde{\theta}_1 = 14.4607$, $\tilde{\theta}_2 = 5.8330$, and $\tilde{\beta} = 3.0581$, respectively. Upon using these initial values, we determine the MLEs, as well as their standard errors (SEs) and the 95% CIs, based on the asymptotic approach and the bootstrap method. These results are presented in Table 35.5. The likelihood ratio test presented in Table 35.6 as well as the CI obtained in Table 35.5 both show that in this case, there is enough evidence against the hypothesis $H_0 : \beta = 1.0$, that is, against the exponential distribution. Once again, we notice here that the SEs from bootstrap are larger than those obtained from the observed Fisher information matrix and also the bootstrap CIs are wider than those determined by the asymptotic approach.

35.7 Concluding Remarks

In this article, we develop inference for coherent systems with known signatures when they are placed on a simple step-stress experiment and that Type-II censored data are available on system lifetimes. We assume the component lifetimes to be distributed as Weibull. It is observed that the maximum likelihood estimates of the model parameters cannot be obtained in a closed-form and a numerical algorithm needs to be employed. MLEs of the model parameters are obtained through direct maximization of the observed likelihood by using the Newton-Raphson method, and construction of asymptotic confidence intervals by using the observed Fisher information matrix and a parametric bootstrap approach is discussed. The detailed steps of the St-EM algorithm for estimating the model parameters under this setup are developed. Within the framework of the St-EM algorithm, the asymptotic confidence intervals for the model

parameters by using the missing information principle are also developed. A method based on BLUEs is proposed for providing initial values for the numerical algorithm. Through a detailed Monte Carlo study, we examine and compare the performances of the two different methods for obtaining point and interval estimates. It is noted that under this setup, the observed likelihood-based direct maximization approach and the St-EM algorithm yield results that are quite close. It is observed that the MLEs are biased, with bias decreasing as sample size increases. It is also observed that the coverage probabilities of the confidence intervals are close to the nominal confidence level in general.

Naturally, it will be of interest to extend these results in different directions, such as for step-stress experiments under time constraint (see Balakrishnan et al. [40]) and step-stress experiment with random stress change times (see Xiong and Milliken [41]). It will also be of interest to determine an optimal time τ for changing the stress level (see Gouno et al. [42]). Work on these problems is currently under progress, and we hope to report the findings in a future paper.

References

1. Nelson, W.B.: Accelerated life testing: step-stress models and data analysis. *IEEE Trans. Reliab.* **29**, 103–108 (1980)
2. Nelson, W.B.: Accelerated Life Testing, Statistical Models, Test Plans and Data Analysis. John Wiley and Sons, New York (1990)
3. Nelson, W.B., Meeker, W.Q.: Theory for optimum accelerated censored life tests for Weibull and extreme value distributed. *Technometrics* **20**, 171–177 (1978)
4. Meeker, W., Hahn, G.: How to Plan Accelerated Life Tests: Some Practical Guidelines. The ASQC Basic References in Quality Control, Wisconsin (1985)
5. Meeker, W.Q., Escobar, L.A.: Statistical Methods for Reliability Data. John Wiley and Sons, New York (1998)
6. Bagdonavicius, V.B., Nikulin, M.: Accelerated Life Models: Modeling and Statistical Analysis. Chapman and Hall CRC Press, Boca Raton, Florida (2002)
7. Miller, R., Nelson, W.B.: Optimum simple step-stress plans for accelerated life testing. *IEEE Trans. Reliab.* **32**, 59–65 (1983)
8. Bai, D.S., Kim, M.S., Lee, S.H.: Optimum simple step-stress accelerated life test with censoring. *IEEE Trans. Reliab.* **38**, 528–532 (1989)

9. DeGroot, M., Goel, P.: Bayesian estimation and optimal design in partially accelerated life testing. *Nav. Res. Logist.* **26**, 223–235 (1979)
10. Balakrishnan, N., Kundu, D., Ng, H.K.T., Kannan, N.: Point and interval estimation for a simple step-stress model with Type-II censoring. *J. Qual. Technol.* **9**, 35–47 (2007)
11. Balakrishnan, N.: A synthesis of exact inferential results for exponential step-stress models and associated optimal accelerated life-tests. *Metrika* **69**, 351–396 (2009)
12. Balakrishnan, N., Xie, Q.: Exact inference for a simple step-stress model with Type-I hybrid censored data from the exponential distribution. *J. Statist. Plann. Inference* **137**, 3268–3290 (2007)
13. Balakrishnan, N., Xie, Q.: Exact inference for a simple step-stress model with Type-II hybrid censored data from the exponential distribution. *J. Statist. Plann. Inference* **137**, 2543–2563 (2007)
14. Han, D., Balakrishnan, N.: Inference for a simple step-stress model with competing risks for failure from the exponential distribution under time constraint. *Comput. Stat. Data Anal.* **54**, 2066–2081 (2010)
15. Kateri, M., Balakrishnan, N.: Inference for a simple step-stress model with Type-II censoring and Weibull distributed lifetime. *IEEE Trans. Reliab.* **57**, 616–626 (2008)
16. Samaniego, F.J.: *System Signatures and Their Applications in Engineering Reliability*. Springer, New York (2007)
17. Navarro, J., Samaniego, F., Balakrishnan, N., Bhattacharya, D.: On the application and extension of system signatures in engineering reliability. *Nav. Res. Logist.* **55**, 313–327 (2008)
18. Balakrishnan, N., Ng, H.K.T., Navarro, J.: Linear inference for Type-II censored lifetime data of reliability systems with known signatures. *IEEE Trans. Reliab.* **60**, 426–440 (2011)
19. Balakrishnan, N., Ng, H.K.T., Navarro, J.: Exact nonparametric inference for component lifetime distribution based on lifetime data from systems with known signatures. *J. Nonparametr. Statist.* **23**, 741–752 (2011)
20. Celeux, G., Diebolt, J.: The SEM algorithm: A probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Comput. Stat. Q.* **2**, 73–82 (1985)
21. Dempster, A., Laird, N., Rubin, D.: Maximum Likelihood from Incomplete Data via the EM Algorithm. *J. R. Stat. Soc. Ser. B* **39**, 1–38 (1977)
22. McLachlan, G., Krishnan, T.: *The EM Algorithm and Extensions*, 2nd edn. John Wiley and Sons, New York (2008)
23. Navarro, J., Ruiz, J., Sandoval, C.: Properties of coherent systems with dependent components. *Commun. Statist. Theory Methods* **36**, 175–191 (2007)
24. Samaniego, F.: On closure of the IFR class under formation of coherent systems. *IEEE Trans. Reliab.* **34**, 69–72 (1985)
25. Kochar, S., Mukerjee, H., Samaniego, F.: The ‘Signature’ of a coherent system and its application to comparisons among systems. *Nav. Res. Logist.* **46**, 507–523 (1999)
26. Arnold, B.C., Balakrishnan, N., Nagaraja, H.N.: *A First Course in Order Statistics*. Society for Industrial and Applied Mathematics, Philadelphia, USA (2008)
27. Ye, Z., Ng, H.: On analysis of incomplete field failure data. *Ann. Appl. Stat.* **8**, 1713–1727 (2014)
28. Celeux, G., Chauveau, D., Diebolt, J.: Stochastic versions of the EM algorithm: an experimental study in the mixture case. *J. Stat. Comput. Simul.* **55**, 287–314 (1996)
29. Chauveau, D.: A stochastic EM algorithm for mixtures with censored data. *J. Statist. Plann. Inference* **46**, 1–25 (1995)
30. Diebolt, J., Celeux, G.: Asymptotic properties of a stochastic EM algorithm for estimating mixing proportions. *Commun. Stat.: Stoch. Model.* **9**, 599–613 (1993)
31. Nielsen, S.: The stochastic EM algorithm: Estimation and asymptotic results. *Bernoulli* **6**, 457–489 (2000)
32. Louis, T.A.: Finding the observed information matrix when using the EM algorithm. **44**, 226–233 (1982)
33. Efron, B., Tibshirani, R.: *An Introduction to the Bootstrap*. Chapman and Hall CRC Press, Boca Raton, Florida (1993)
34. Hamada, M.S., Wilson, A., Reese, C.S., Martz, H.: *Bayesian Reliability*. Springer, New York (2008)
35. Ganguly, A., Kundu, D., Mitra, S.: Bayesian Analysis of Simple Step-stress Model under Weibull Lifetimes. *IEEE Trans. Reliab.* **64**, 473–485 (2015)
36. Jablonka, A., Cramer, E., Hermanns, M.: Statistical inference for coherent systems with Weibull distributed component lifetimes under complete and incomplete information. *Appl. Stoch. Model. Bus. Ind.* (2019). <https://doi.org/10.1002/asmb.2440>
37. Berger, J.O., Sun, D.: Bayesian analysis for the Poly-Weibull distribution. *J. Am. Stat. Assoc.* **88**, 1412–1418 (1993)
38. Kundu, D.: Bayesian inference and life testing plan for the Weibull distribution in presence of progressive censoring. *Technometrics* **50**, 144–154 (2008)
39. Lindley, D.V.: Approximate Bayes method. *Trab. Estad.* **31**, 223–237 (1980)
40. Balakrishnan, N., Xie, Q., Kundu, D.: Exact inference for a simple step-stress model from the exponential distribution under time constraint. *Ann. Inst. Stat. Math.* **61**, 251–274 (2009)
41. Xiong, C., Milliken, G.A.: Step-stress life testing with random stress changing times for exponential data. *IEEE Trans. Reliab.* **48**, 141–148 (1999)
42. Gouno, E., Sen, A., Balakrishnan, N.: Optimal step-stress test under progressive Type-I censoring. *IEEE Trans. Reliab.* **53**, 388–393 (2004)



Dr. N. Balakrishnan is a Distinguished University Professor in the Department of Mathematics and Statistics at McMaster University, Hamilton, Ontario, Canada. He is a Fellow of the American Statistical Association, Fellow of the Institute of Mathematical Statistics and an Elected Member of the International Statistical Institute. He has received an Honorary Doctorate Degree in 2017 from The National University of Athens, Greece. He has varied research interests that include Ordered Data Analysis, Reliability Theory, Survival Inference, Censoring Methodology, and Univariate and Multivariate Distribution Theory. He is currently the Editor-in-Chief of *Communications in Statistics* (three series) published by Taylor and Francis, Philadelphia.



Debanjan Mitra received his PhD from McMaster University, Canada in 2012. He was an Assistant Professor in the Department of Mathematics, Indian Institute of Technology Guwahati. Currently, he is an Assistant Professor in Operations Management, Quantitative Methods and Information Systems Area at Indian Institute of Management Udaipur. His current research interests include lifetime data analysis, censoring methodologies, statistical quality control etc.



Dr. Xiaojun Zhu is a Lecturer in the Department of Mathematical Sciences at Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, China. He received his PhD in Statistics from McMaster University, Ontario, Canada, in 2015. His research interests include distribution theory, order Statistics, reliability, survival analysis and non-parametric inference.



Bivariate Distributions with Singular Components

Debasis Kundu

Contents

36.1	Introduction	733
36.2	Preliminaries	735
36.2.1	Proportional Hazard Class.....	735
36.2.2	Proportional Reversed Hazard Class.....	736
36.2.3	Copula.....	736
36.2.4	Three Important Distributions.....	736
36.3	Two Main Approaches	737
36.3.1	Minimization Approach (Model 1).....	737
36.3.2	Maximization Approach (Model 2).....	739
36.4	Some Special Cases	740
36.4.1	Model 1.....	740
36.4.2	Model 2.....	746
36.5	Classical Inference	746
36.5.1	EM Algorithm: Model 1.....	748
36.5.2	EM Algorithm: Model 2.....	750
36.6	Data Analysis	752
36.7	Some Other Bivariate Distributions	753
36.7.1	Sarhan-Balakrishnan Bivariate Distribution.....	753
36.7.2	Modified Sarhan-Balakrishnan Bivariate Distribution.....	754
36.7.3	Bivariate Weibull-Geometric Distribution.....	755
36.7.4	Bivariate PHM-Geometric Distribution.....	756
36.7.5	Bivariate GE-Geometric Distribution.....	758
36.8	Conclusions	759
References	759

Abstract

In this chapter we mainly discuss classes of bivariate distributions with singular components. It is observed that there are mainly two different ways of defining bivariate distributions with singular components, when the marginals are absolutely continuous. Most of the bivariate

distributions available in the literature can be obtained from these two general classes. A connection between the two approaches can be established based on their copulas. It is observed that under certain restrictions both these classes have very similar copulas. Several properties can be established of these proposed classes. It is observed that the maximum likelihood estimators (MLEs) may not always exist; whenever they exist, they cannot be obtained in closed forms. Numerical techniques are needed to compute the MLEs of the unknown parameters. Alternatively, very efficient expectation maximization (EM) algorithm can be used to compute the MLEs. The corresponding observed Fisher information matrix also can be obtained quite conveniently at the last stage of the EM algorithm, and it can be used to construct confidence intervals of the unknown parameters. The analysis of one data set has been performed to see the effectiveness of the EM algorithm. We discuss different generalizations, propose several open problems, and finally conclude the chapter.

Keywords

Absolute continuous distribution · Singular distribution · Fisher information matrix · EM algorithm · Joint probability distribution function · Joint probability density function

AMS Subject Classifications: 62F10, 62F03, 62H12

36.1 Introduction

Bivariate continuous distributions occur quite naturally in practice. An extensive amount of work has been done on different bivariate continuous distributions in the statistical literature. Some of the well-known absolutely continuous

D. Kundu (✉)
Department of Mathematics and Statistics, Indian Institute of Technology Kanpur, Kanpur, Uttar Pradesh, India
e-mail: kundu@iitk.ac.in

bivariate continuous distributions are bivariate normal, bivariate- t , bivariate log-normal, bivariate gamma, bivariate extreme value, bivariate Birnbaum-Saunders distributions, bivariate skew normal distribution, bivariate geometric skew normal distribution, etc., see, for example, the books by Balakrishnan and Lai [7], Kotz et al. [27] on different bivariate and multivariate distributions, the recent review article by Balakrishnan and Kundu [6] on bivariate and multivariate Birnbaum-Saunders distributions, the article by Azzalini and Dalla Valle [4] and the monograph by Azzalini and Capitanio [3] on multivariate skew-normal distribution, the recent article by Kundu [29] on multivariate geometric skew-normal distribution, and the references cited therein. The main purpose of any bivariate distribution is to model the two marginals and also to find association between the two marginals.

It may be mentioned that although there are numerous absolutely continuous bivariate distributions available in the literature, if there are ties in the data set, then these absolutely continuous bivariate distributions cannot be used to analyze this data set. Sometimes, the ties may occur due to truncation, but in many situations the ties may occur naturally and with a positive probability. To analyze a bivariate data set with ties, one needs a bivariate model with a singular component. These class of bivariate distributions assign a positive probability on $X = Y$, where X and Y denote the two marginals, and both are assumed to be absolutely continuous.

Marshall and Olkin [44] first proposed a bivariate distribution such that its both the marginals X and Y have exponential distributions, and $P(X = Y) > 0$. From now on we call this distribution as the Marshall-Olkin bivariate exponential (MOBE) distribution, and popularly it is also known as the shock model. Since its inception, an extensive amount of work has been done related to this distribution. Several properties have been established, its characterizations, and both classical and Bayesian inferential procedures have been developed, see, for example, Arnold [2], Baxter and Rachev [8], Boland [14], Muliere and Scarsini [50], Pena and Gupta [54], Ryu [57], the review article by Nadarajah [51], and the references cited therein.

Lu [41] provided the Weibull extension of the MOBE model. Since then quite a bit of work has been done on this and some related distributions mainly developing inference procedures under complete sample and under various sampling schemes and develop the properties of the order statistics. This model has been used quite successfully to analyze dependent competing risks data. See, for example, Begum and Khan [10], Cai et al. [15], Feizjadian and Hashemi [18], Jose et al. [25], Kundu and Dey [30], Kundu and Gupta [35], and Lai et al. [39], and see the references cited therein.

Some of the other bivariate distributions with singular components which can be found in the literature are bivariate Kumaraswamy (BVK), bivariate Pareto (BVP), bivariate double generalized exponential (BDGE), bivariate exponentiated Frechet (BEF), and bivariate Gumbel (BVG) distributions; see, for example, Barreto-Souza and Lemonte [9], bivariate generalized exponential (BGE) model of Kundu and Gupta [32], Sarhan-Balakrishnan's bivariate (SBB) distribution introduced by Sarhan and Balakrishnan [58], modified Sarhan-Balakrishnan's bivariate (MSBB) distribution introduced by Kundu and Gupta [33], bivariate model with proportional reversed hazard marginals proposed by Kundu and Gupta [34], bivariate generalized linear failure rate model introduced by Sarhan et al. [60], the generalized Marshall-Olkin bivariate distributions introduced by Gupta et al. [23], and bivariate inverse Weibull distribution as proposed by Muhammed [49] and Kundu and Gupta [37], and see the references cited therein.

In many situations although the data are continuous in nature, say time, pressure, etc., they are often measured in discrete units. In a situation like this, we often get ties in a bivariate data set. But we will provide few examples where ties occur naturally.

Shock Model It was originally proposed by Marshall and Olkin [44], and it is considered to be the most classical model of a bivariate distribution with a singular component. Suppose there are two components of a system, and there are three shocks which can affect the two components. The shocks appear randomly, and they affect the systems. Shock 1 affects the Component 1, Shock 2 affects the Component 2, and Shock 3 affects both the components. The component fails as soon as it receives a shock. The failure times of the both the components are observed as a bivariate random variable. In this case clearly there is a positive probability that the failure times of the two components become equal.

Stress Model It was originally proposed by Kundu and Gupta [32], and it can be described as follows. Suppose a system has two components, and each component is subject to individual stress, say V_1 and V_2 . Other than the individual stresses, the system has an overall stress V_3 which has been propagated to both the components equally irrespective of their individual stresses. Therefore, the observed stress at the two components are $X = \max\{V_1, V_3\}$ and $Y = \max\{V_2, V_3\}$, respectively.

Soccer Model Suppose the first component of a bivariate data represents the time of the first kick goal scored by any team, and second component represents the time of the first goal of any type by the home team. In this case also if the first goal is scored by the home team and it is a kick goal,

then there is a tie of the two components, and it happens with a positive probability.

Maintenance Model Suppose a system has two components, say Component 1 and Component 2, and it is assumed that both components have been maintained independently, and also there is an overall maintenance to both the components. It is assumed that due to component maintenance, suppose the lifetime of Component i is increased by U_i amount, for $i = 1$ and 2, and because of the overall maintenance, the lifetime of each component is increased by U_3 amount. Hence, the increased lifetimes of the two components are $X_1 = \max\{U_1, U_3\}$ and $X_2 = \max\{U_2, U_3\}$, respectively, see, for example, Kundu and Gupta [32] in this respect.

Note that most of the bivariate distributions with singular components are based on two different approaches, namely, minimization approach and maximization approach. The main aim of this manuscript is to put both the methods under the same framework. It may be mentioned that any bivariate distribution is characterized by its marginals and the copula function. It is observed that based on the copula many properties can be derived for any class of bivariate distribution functions. We derive some basic properties in both the cases, and it is observed that under certain restrictions they can be generated from very similar copulas. Some specific examples have been provided.

The maximum likelihood estimators (MLEs) of the unknown parameters may not always exist, and even if they exist, they cannot be obtained in explicit forms. One needs to solve a higher dimensional optimization problem to compute the MLEs. To avoid that we have proposed to use this problem as a missing value problem, and we have used a very efficient EM algorithm to compute the MLEs of the unknown parameters. It avoids solving a higher-dimensional optimization problem. Moreover, the observed Fisher information matrix also can be obtained quite conveniently at the last step of the EM algorithm, and it can be used to construct confidence intervals of the unknown parameters. The analysis of one real-life data set has been performed to see the effectiveness of the EM algorithm.

Finally, we provide few examples, and they cannot be obtained directly using the two methods which we have mentioned above. In all the cases we have provided explicit expressions of the joint PDF and have mentioned the estimation procedures of the unknown parameters in each case. We have further mentioned few bivariate distributions with singular components which cannot be obtained by the above two methods, and finally we conclude the paper.

The rest of the paper is organized as follows. In Sect. 36.2, we provide some preliminaries, and in Sect. 36.3 we describe the two main approaches to produce bivariate distribution with a singular component. Some special cases are presented

in Sect. 36.4, and the MLEs are provided in Sect. 36.5. In Sect. 36.6 we have provided the analysis of a data set. We have provided examples of few bivariate distributions which cannot be obtained by the proposed methods in Sect. 36.7, and finally we presented several open problems and conclude the paper in Sect. 36.8.

36.2 Preliminaries

In this section we discuss two important class of distribution functions, namely, (i) proportional hazard class and (ii) proportional reversed hazard class of distribution functions. We will also discuss briefly about the copula function and three important class of distribution functions which will be used quite extensively later.

36.2.1 Proportional Hazard Class

Suppose $F_B(t; \theta)$ is a distribution with the support on the positive real axis as mentioned before and $S_B(t, \theta) = 1 - F_B(t; \theta)$ is the corresponding survival function. Let us consider the class of distribution functions which has the survival function of the following form:

$$S_{PHM}(t; \alpha, \theta) = [S_B(t; \theta)]^\alpha; \quad t > 0, \quad (36.1)$$

with parameters $\theta, \alpha > 0$, and zero otherwise. Here θ can be vector valued, and $S_B(t; \theta)$ is called as the base line survival function. In this case the class of distribution functions defined by (36.1) is known as the proportional hazard model (PHM). In this case the PDF of the PHM becomes

$$f_{PHM}(t; \alpha, \theta) = \alpha f_B(t; \theta) [S_B(t; \theta)]^{\alpha-1}; \quad t \geq 0, \quad (36.2)$$

and zero otherwise. The proportional hazard model was originally proposed by Cox [16] as a regression model in the life-table data analysis. The class of distribution functions defined through the survival function (36.1) is called the proportional hazard class because if the hazard function of $f_B(t; \theta)$ is

$$h_B(t; \theta) = \frac{f_B(t; \theta)}{S_B(t; \theta)},$$

then the hazard function of $f_{PHM}(t; \alpha, \theta)$ becomes

$$h_{PHM}(t; \alpha, \theta) = \frac{f_{PHM}(t; \alpha, \theta)}{S_{PHM}(t; \alpha, \theta)} = \alpha \frac{f_B(t; \theta)}{S_B(t; \theta)} = \alpha h_B(t; \theta).$$

Hence, in this case the hazard function of any member of the proportional hazard class is proportional to the base line hazard function. Since the inception of the model by

Cox [16], an extensive amount of work has been done related to Cox's PHMs. Most of the standard statistical books on survival analysis discuss this model in detail; see, for example, Cox and Oakes [17], Therneau and Grambsch [63], and the references cited therein.

36.2.2 Proportional Reversed Hazard Class

Suppose $F_B(t; \theta)$ is a distribution with the support on the positive real axis. Then consider the class of distribution functions of the form

$$F_{PRHM}(t; \alpha, \theta) = [F_B(t; \theta)]^\alpha; \quad t > 0, \quad (36.3)$$

with parameters $\alpha > 0$ and θ (may be a vector valued) and base line distribution function $F_B(t; \theta)$. This class of distribution functions is known as the proportional reversed hazard model (PRHM). If $F_B(t; \theta)$ admits the PDF $f_B(t; \theta)$, then the PRHM has a PDF

$$f_{PRHM}(t; \alpha, \theta) = \alpha [F_B(t; \theta)]^{\alpha-1} f_B(t; \theta); \quad t \geq 0.$$

Lehmann [43] first proposed this model in the context of hypotheses testing. It is known as the proportional reversed hazard class because if the base line distribution function $F_B(t; \theta)$ has the reversed hazard function

$$r_B(t; \theta) = \frac{f_B(t; \theta)}{F_B(t; \theta)},$$

then $F_{PRHM}(t; \alpha, \theta)$ has the reversed hazard function

$$r_{PRHM}(t; \alpha, \theta) = \frac{f_{PRHM}(t; \alpha, \theta)}{F_{PRHM}(t; \alpha, \theta)} = \alpha \frac{f_B(t; \theta)}{F_B(t; \theta)} = \alpha r_B(t; \theta).$$

Hence, the reversed hazard function of any member of the proportional reversed hazard class is proportional to the base line reversed hazard function. For a detailed discussion on this issue, one is referred to Block et al. [13]. An extensive amount of work has been done on different proportional reversed hazard classes; see, for example, exponentiated Weibull distribution of Mudholkar et al. [48], generalized exponential distribution of Gupta and Kundu [21], exponentiated Rayleigh of Surles and Padgett [62], and generalized linear failure rate model of Sarhan and Kundu [59], see also Kundu and Gupta [31] and the references cited therein.

36.2.3 Copula

The dependence between two random variables, say X and Y , is completely described by the joint distribution function

$F_{X,Y}(x, y)$. The main idea of separating $F_{X,Y}(x, y)$ in two parts, the one which describes the dependence structure, and the other one which describes the marginal behavior, leads to the concept of copula. To every bivariate distribution function $F_{X,Y}(x, y)$, with continuous marginals $F_X(x)$ and $F_Y(y)$, corresponds to a unique function $C : [0, 1] \times [0, 1] \rightarrow [0, 1]$, called a copula function such that

$$F_{X,Y}(x, y) = C(F_X(x), F_Y(y)); \quad \text{for } (x, y) \in (-\infty, \infty) \times (-\infty, \infty).$$

Note that $C(u, v)$ is a proper distribution function on $[0, 1] \times [0, 1]$. Moreover, from Sklar's theorem (see, e.g., Nelsen [53]), it follows that if $F_{X,Y}(\cdot, \cdot)$ is a joint distribution function with continuous marginals $F_X(\cdot)$, $F_Y(\cdot)$, and if $F_X^{-1}(\cdot)$, $F_Y^{-1}(\cdot)$ are the inverse functions of $F_X(\cdot)$, $F_Y(\cdot)$, respectively, then there exists a unique copula C in $[0, 1] \times [0, 1]$, such that

$$C(u, v) = F_{X,Y}(F_X^{-1}(u), F_Y^{-1}(v)); \quad \text{for } (u, v) \in [0, 1] \times [0, 1].$$

Moreover, if $S_{X,Y}(x, y)$ is the joint survival function of X and Y , and $S_X(x)$ and $S_Y(y)$ are survival functions of X and Y , respectively, then there exists unique function $\bar{C} : [0, 1] \times [0, 1] \rightarrow [0, 1]$, called a (survival) copula function such that

$$S_{X,Y}(x, y) = \bar{C}(S_X(x), S_Y(y)); \quad \text{for } (x, y) \in (-\infty, \infty) \times (-\infty, \infty).$$

In this case

$$\bar{C}(u, v) = S_{X,Y}(S_X^{-1}(u), S_Y^{-1}(v)); \quad \text{for } (u, v) \in [0, 1] \times [0, 1].$$

Moreover,

$$\bar{C}(u, v) = u + v - 1 + C(1 - u, 1 - v); \quad \text{for } (u, v) \in [0, 1] \times [0, 1].$$

It should be pointed out that the survival copula is also a copula, i.e., $\bar{C}(u, v)$ is also a proper distribution function on $[0, 1] \times [0, 1]$. It is well known that many dependence properties of a bivariate distribution are copula properties and, therefore, can be obtained by studying the corresponding copula. These properties do not depend on the marginals.

36.2.4 Three Important Distributions

In this section we discuss three important distribution functions which will be used quite extensively in our future development.

Exponential Distribution

A random variable X is said to have an exponential distribution with the parameter $\lambda > 0$, if the CDF of X is as follows:

$$F_X(x; \lambda) = P(X \leq x) = 1 - e^{-\lambda x}; \quad x > 0,$$

and zero, otherwise. The corresponding PDF of X becomes

$$f_X(x; \lambda) = \lambda e^{-\lambda x}; \quad \text{for } x > 0,$$

and zero, otherwise. From now on we will denote it by $\text{Exp}(\lambda)$. The PDF of an exponential distribution is always a decreasing function, and it has a constant hazard function for all values of λ . The exponential distribution is the most used distribution in lifetime data analysis. It has several interesting properties including the lack of memory property, and it belongs to the PHM class. Interested readers are referred to Balakrishnan and Basu [5] for a detailed discussions on exponential distribution.

Weibull Distribution

A random variable X is said to have a two-parameter Weibull distribution if it has the following CDF:

$$F_X(x; \alpha, \lambda) = 1 - e^{-\lambda x^\alpha}; \quad \text{for } x > 0,$$

and zero, otherwise. Here, $\alpha > 0$ is called the shape parameter and $\lambda > 0$ as the scale parameter. The corresponding PDF becomes

$$f_X(x; \alpha, \lambda) = \alpha \lambda x^{\alpha-1} e^{-\lambda x^\alpha}; \quad \text{for } x > 0,$$

and zero, otherwise. From now on it will be denoted by $\text{WE}(\alpha, \lambda)$.

A two-parameter Weibull distribution is more flexible than a one-parameter exponential distribution. The shape of the PDF and the hazard function depend on the shape parameter α . The PDF can be a decreasing or a unimodal function if $\alpha \leq 1$ or $\alpha > 1$, respectively. Similarly, for $\alpha \leq 1$, the hazard function is a decreasing function, and for $\alpha > 1$, the hazard function is an increasing function. Because of its flexibility it has been used quite extensively in reliability and in survival analysis. It is a PHM. An excellent handbook on Weibull distribution is by Rinne [56]. Interested readers are referred to that handbook for further reading.

Generalized Exponential Distribution

A random variable X is said to have a two-parameter generalized exponential distribution if it has the following CDF:

$$F_X(x; \alpha, \lambda) = (1 - e^{-\lambda x})^\alpha; \quad \text{for } x > 0,$$

and zero, otherwise. Here also $\alpha > 0$ is called the shape parameter and $\lambda > 0$ as the scale parameter. The corresponding PDF becomes

$$f_X(x; \alpha, \lambda) = \alpha \lambda e^{-\lambda x} (1 - e^{-\lambda x})^{\alpha-1}; \quad \text{for } x > 0,$$

and zero, otherwise. From now on we will denote it by $\text{GE}(\alpha, \lambda)$.

A two-parameter GE distribution was first introduced by Gupta and Kundu [21] as an alternative to the two-parameter Weibull and gamma distribution. Since it has been introduced, it has been used quite extensively in analyzing different lifetime data. It may be mentioned that it is a PRHM. Interested readers are referred to the review article by Nadarajah [52] or a book length treatment by Al-Hussaini and Ahsanullah [1] for different developments on the GE distribution till date.

36.3 Two Main Approaches

In this section we provide the two main approaches, namely, the minimization and maximization approaches, to construct a bivariate distribution with a singular component. We provide both the methods briefly and discuss several common properties of the general class of distribution functions. It is assumed throughout that $F_B(t; \theta)$ is an absolutely continuous distribution function with the support on the positive real axis and $S_B(t; \theta) = 1 - F_B(t; \theta)$. Moreover, the PDF of $F_B(t; \theta)$ is $f_B(t; \theta)$ for $t > 0$ and zero, otherwise. Here θ can be vector valued also as mentioned in the previous section.

36.3.1 Minimization Approach (Model 1)

In this section we provide the bivariate distributions with singular components which are based on minimum. Suppose U_1, U_2 , and U_3 are three independent nonnegative random variables with survival functions $S_1(t; \alpha_1, \theta) = [S_B(t; \theta)]^{\alpha_1}$, $S_2(t; \alpha_2, \theta) = [S_B(t; \theta)]^{\alpha_2}$, $S_3(t; \alpha_3, \theta) = [S_B(t; \theta)]^{\alpha_3}$, respectively for $\alpha_1 > 0, \alpha_2 > 0, \alpha_3 > 0$. Now we define a new bivariate random variable (X, Y) as follows:

$$X = \min\{U_1, U_3\} \quad \text{and} \quad Y = \min\{U_2, U_3\}. \quad (36.4)$$

Note that although U_1, U_2 , and U_3 are independent, due to presence of U_3 in both X and Y , X and Y are dependent. We would like to obtain the joint cumulative distribution function (JCDF) and the joint probability density function (JPDF) of X and Y . But before that let us observe the following facts. Since X is defined as in (36.4), the survival function of X becomes

$$\begin{aligned} P(X > x) &= S_X(x; \alpha_1, \alpha_3, \theta) = P(U_1 > x, U_3 > x) \\ &= P(U_1 > x)P(U_3 > x) = [S_B(x; \theta)]^{\alpha_1 + \alpha_3}. \end{aligned}$$

Hence, the survival function and the CDF of X depend on θ and $\alpha_1 + \alpha_3$. Similarly, the survival function of Y becomes

$P(Y > y) = [S_B(y; \theta)]^{\alpha_2 + \alpha_3}$. Hence, if U_1, U_2 , and U_3 are absolutely continuous random variables, then X and Y are also absolutely continuous random variables. Moreover, in this case

$$\begin{aligned} P(X = Y) &= P(U_3 < U_1, U_3 < U_2) \\ &= \alpha_3 \int_0^\infty f_B(t; \theta) [S_B(t; \theta)]^{\alpha_3 - 1} [S_B(t; \theta)]^{\alpha_1} [S_B(x; \theta)]^{\alpha_2} dt \\ &= \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} \int_0^\infty (\alpha_1 + \alpha_2 + \alpha_3) f_B(t; \theta) [S_B(t; \theta)]^{\alpha_1 + \alpha_2 + \alpha_3 - 1} \\ &= \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} > 0. \end{aligned} \quad (36.5)$$

Hence, (36.5) indicates that $X = Y$ has a positive probability and for fixed α_1 and α_2 , $\lim_{\alpha_3 \rightarrow 0} P(X = Y) = 0$ and $\lim_{\alpha_3 \rightarrow \infty} P(X = Y) = 1$. Along the same line it can be easily obtained that

$$P(X < Y) = \frac{\alpha_1}{\alpha_1 + \alpha_2 + \alpha_3} \quad \text{and} \quad P(Y < X) = \frac{\alpha_2}{\alpha_1 + \alpha_2 + \alpha_3}. \quad (36.6)$$

Now we will provide the joint survival function of X and Y and also derive the joint PDF of X and Y . The joint survival function of X and Y can be written as

$$\begin{aligned} S_{X,Y}(x, y) &= P(X > x, Y > y) \\ &= P(U_1 > x, U_2 > y, U_3 > \max\{x, y\}) \\ &= [S_B(x, \theta)]^{\alpha_1} [S_B(y, \theta)]^{\alpha_2} [S_B(z, \theta)]^{\alpha_3}, \end{aligned} \quad (36.7)$$

here $z = \max\{x, y\}$. Equivalently, (36.7) can be written as follows:

$$f(x, y) = \begin{cases} \alpha_1(\alpha_2 + \alpha_3) f_B(x, \theta) [S_B(x, \theta)]^{\alpha_1 - 1} f_B(y, \theta) [S_B(y, \theta)]^{\alpha_2 + \alpha_3 - 1} & \text{if } x < y \\ \alpha_2(\alpha_1 + \alpha_3) f_B(x, \theta) [S_B(x, \theta)]^{\alpha_1 + \alpha_3 - 1} f_B(y, \theta) [S_B(y, \theta)]^{\alpha_2 - 1} & \text{if } x > y. \end{cases}$$

Now it can be easily observed that

$$\int_0^\infty \int_0^\infty f(x, y) dx dy = \int_0^\infty \int_y^\infty f(x, y) dx dy + \int_0^\infty \int_x^\infty f(x, y) dy dx = \frac{\alpha_1 + \alpha_2}{\alpha_1 + \alpha_2 + \alpha_3} < 1.$$

Hence, clearly $S_{X,Y}(x, y)$ is not an absolutely continuous survival function. Moreover,

$$\int_0^\infty \int_0^\infty f_{X,Y}(x, y) dx dy + P(X = Y) = 1.$$

$$S_{X,Y}(x, y) = \begin{cases} [S_B(x, \theta)]^{\alpha_1} [S_B(y, \theta)]^{\alpha_2 + \alpha_3} & \text{if } x < y \\ [S_B(x, \theta)]^{\alpha_1 + \alpha_3} [S_B(y, \theta)]^{\alpha_2} & \text{if } x \geq y. \end{cases} \quad (36.8)$$

Now we will show that the joint survival function (36.7) or (36.8) is not an absolutely continuous survival function. Let us recall that a joint survival function $S(x, y)$ is said to be absolute continuous if there exists a $f(x, y) \geq 0$, such that

$$S(x, y) = \int_x^\infty \int_y^\infty f(u, v) du dv \quad \text{for all } x > 0, y > 0.$$

In that case $f(x, y)$ can be recovered from $S(x, y)$ as

$$f(x, y) = \frac{\partial^2}{\partial x \partial y} S(x, y).$$

Let us denote $f(x, y) = \frac{\partial^2}{\partial x \partial y} S_{X,Y}(x, y)$. Hence, from (36.8),

Let us denote for $x > 0$ and $y > 0$

$$f_{ac}(x, y) = \frac{\alpha_1 + \alpha_2 + \alpha_3}{\alpha_1 + \alpha_2} f(x, y). \tag{36.9}$$

Clearly, $f_{ac}(x, y)$ is a bivariate density function on the positive quadrant. Observe that for $x > 0, y > 0$ and for $z = \max\{x, y\}$,

$$\begin{aligned} S_{X,Y}(x, y) &= P(X > x, Y > y) = \int_x^\infty \int_y^\infty f(u, v) du dv + P(X = Y > z) \\ &= \frac{\alpha_1 + \alpha_2}{\alpha_1 + \alpha_2 + \alpha_3} \int_x^\infty \int_y^\infty f_{ac}(u, v) du dv + \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} \int_z^\infty f_s(u) du. \end{aligned} \tag{36.10}$$

Here $f_{ac}(u, v)$ is same as defined in (36.9) and

$$f_s(u) = (\alpha_1 + \alpha_2 + \alpha_3) f_B(u; \theta) [S_B(u; \theta)]^{\alpha_1 + \alpha_2 + \alpha_3 - 1},$$

and it is a probability density function on the positive real axis. Based on (36.10), for $x > 0, y > 0$ and for $z = \max\{x, y\}$, the random variable (X, Y) has the joint PDF $f_{X,Y}(x, y)$ of the form

$$f_{X,Y}(x, y) = \begin{cases} \alpha_1(\alpha_2 + \alpha_3) f_B(x, \theta) [S_B(x, \theta)]^{\alpha_1 - 1} f_B(y, \theta) [S_B(y, \theta)]^{\alpha_2 + \alpha_3 - 1} & \text{if } x < y \\ \alpha_2(\alpha_1 + \alpha_3) f_B(x, \theta) [S_B(x, \theta)]^{\alpha_1 + \alpha_3 - 1} f_B(y, \theta) [S_B(y, \theta)]^{\alpha_2 - 1} & \text{if } x > y \\ \alpha_3 f_B(x; \theta) [S_B(x; \theta)]^{\alpha_1 + \alpha_2 + \alpha_3 - 1} & \text{if } x = y. \end{cases} \tag{36.11}$$

In this case the random variable (X, Y) has an absolute continuous part and a singular part. The function $f_{X,Y}(x, y)$ is considered to be a density function of (X, Y) , if it is understood that the first two terms are densities with respect to a two-dimensional Lebesgue measure and the third term is a density function with respect to a one dimensional Lebesgue measure; see, for example, Bemis, Bain, and Higgins [11]. It simply means that

$$P(X > x, Y > y) = \int_x^\infty \int_y^\infty f_{X,Y}(u, v) du dv + \int_{\max\{x,y\}}^\infty f_{X,Y}(v, v) dv.$$

From (36.10) it is immediate that

$$S_{X,Y}(x, y) = \frac{\alpha_1 + \alpha_2}{\alpha_1 + \alpha_2 + \alpha_3} S_{ac}(x, y) + \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} S_{si}(x, y), \tag{36.12}$$

here $S_{ac}(x, y)$ is the absolutely continuous part of the survival function,

$$S_{ac}(x, y) = \int_x^\infty \int_y^\infty f_{ac}(u, v) du dv,$$

and $S_{si}(x, y)$ is the singular component of $S_{X,Y}(x, y)$, and it can be written for $z = \max\{x, y\}$, as

$$S_{si}(x, y) = [S_B(z; \theta)]^{\alpha_1 + \alpha_2 + \alpha_3}.$$

Now we would like to find the copula associated with $S_{X,Y}(x, y)$. Since X and Y have the survival functions as $[S_B(x; \theta)]^{\alpha_1 + \alpha_3}$ and $[S_B(y; \theta)]^{\alpha_2 + \alpha_3}$, respectively, therefore

$$\bar{C}(u, v) = \begin{cases} u^{\frac{\alpha_1}{\alpha_1 + \alpha_3}} v & \text{if } u < v^{\frac{\alpha_1 + \alpha_3}{\alpha_2 + \alpha_3}} \\ u v^{\frac{\alpha_2}{\alpha_2 + \alpha_3}} & \text{if } u \geq v^{\frac{\alpha_1 + \alpha_3}{\alpha_2 + \alpha_3}}. \end{cases}$$

If we write $\beta = \alpha_3/(\alpha_1 + \alpha_3)$ and $\delta = \alpha_3/(\alpha_2 + \alpha_3)$, then

$$\bar{C}(u, v) = \begin{cases} u^{1-\beta} v & \text{if } u^\beta < v^\delta \\ u v^{1-\delta} & \text{if } u^\beta \geq v^\delta. \end{cases}$$

If we consider a special case $\alpha_1 = \alpha_2$, and $\eta = \alpha_3/(\alpha_1 + \alpha_3) = \alpha_3/(\alpha_2 + \alpha_3)$, then

$$\bar{C}(u, v) = \begin{cases} u^{1-\eta} v & \text{if } u < v \\ u v^{1-\eta} & \text{if } u \geq v. \end{cases}$$

36.3.2 Maximization Approach (Model 2)

In the last section we had provided the bivariate distributions with singular components which are based on minimum, and in this section we provide the bivariate distributions with singular components which are based on maximum. Let us assume that V_1, V_2 , and V_3 are three independent nonnegative random variables with distribution functions $F_1(t; \beta_1, \lambda) =$

$[F_B(t; \lambda)]^{\beta_1}$, $F_2(t; \beta_2, \lambda) = [F_B(t; \lambda)]^{\beta_2}$, and $F_3(t; \beta_3, \lambda) = [F_B(t; \lambda)]^{\beta_3}$, respectively, for $\beta_1 > 0$, $\beta_2 > 0$, and $\beta_3 > 0$. Let us define a new bivariate random variable (X, Y) as follows:

$$X = \max\{V_1, V_3\} \quad \text{and} \quad Y = \max\{V_2, V_3\}. \quad (36.13)$$

In this case also similarly as before, X and Y will be dependent random variables due to the presence of V_3 . The following results can be easily obtained following the same line as the previous section. The CDFs of X and Y become

$$P(X \leq x) = F_X(x; \beta_1, \beta_3, \lambda) = [F_B(x; \lambda)]^{\beta_1 + \beta_3} \quad \text{and}$$

$$P(Y \leq y) = F_Y(y; \beta_2, \beta_3, \lambda) = [F_B(y; \lambda)]^{\beta_2 + \beta_3}.$$

$$P(X = Y) = \frac{\beta_3}{\beta_1 + \beta_2 + \beta_3}, \quad P(X < Y) = \frac{\beta_1}{\beta_1 + \beta_2 + \beta_3}, \quad P(X > Y) = \frac{\beta_2}{\beta_1 + \beta_2 + \beta_3}.$$

The joint CDF of X and Y for $z = \min\{x, y\}$ becomes

$$F_{X,Y}(x, y) = P(X \leq x, Y \leq y)$$

$$= P(V_1 \leq x, V_2 \leq y, V_3 \leq \min\{x, y\})$$

$$= [F_B(x, \lambda)]^{\beta_1} [F_B(y, \lambda)]^{\beta_2} [F_B(z, \lambda)]^{\beta_3}$$

$$= \begin{cases} [F_B(x, \lambda)]^{\beta_1 + \beta_3} [F_B(y, \lambda)]^{\beta_2} & \text{if } x < y \\ [F_B(x, \lambda)]^{\beta_2 + \beta_3} [F_B(x, \lambda)]^{\beta_1} & \text{if } x \geq y \end{cases} \quad (36.14)$$

Following the same approach as before, the joint PDF of X and Y can be obtained as

$$f_{X,Y}(x, y) = \begin{cases} \beta_2(\beta_1 + \beta_3)f_B(x, \lambda)[F_B(x, \lambda)]^{\beta_1 + \beta_3 - 1}f_B(y, \lambda)[F_B(y, \lambda)]^{\beta_2 - 1} & \text{if } x < y \\ \beta_1(\beta_2 + \beta_3)f_B(x, \lambda)[S_B(x, \theta)]^{\beta_1 - 1}f_B(y, \theta)[F_B(y, \theta)]^{\beta_2 + \beta_3 - 1} & \text{if } x > y \\ \beta_3f_B(x; \lambda)[F_B(x; \lambda)]^{\beta_1 + \beta_2 + \beta_3 - 1} & \text{if } x = y. \end{cases} \quad (36.15)$$

In this case also the joint CDF of the random variable (X, Y) has an absolute continuous part and a singular part, and the function (36.15) is considered to be the joint PDF of X and Y in the same sense as before. Moreover, the copula associate with the joint CDF $F_{X,Y}(x, y)$ becomes

$$C(u, v) = \begin{cases} uv^{\frac{\beta_2}{\beta_2 + \beta_3}} & \text{if } u < v^{\frac{\beta_1 + \beta_3}{\beta_2 + \beta_3}} \\ u^{\frac{\beta_1}{\beta_1 + \beta_3}}v & \text{if } u \geq v^{\frac{\beta_1 + \beta_3}{\beta_2 + \beta_3}} \end{cases} \quad (36.16)$$

Therefore, if we write as before that $\beta = \beta_3/(\beta_1 + \beta_3)$ and $\delta = \beta_3/(\beta_2 + \beta_3)$, then (36.16) becomes

$$C(u, v) = \begin{cases} uv^{1 - \delta} & \text{if } u^\beta < v^\delta \\ u^{1 - \beta}v & \text{if } u^\beta \geq v^\delta. \end{cases}$$

Hence, for the special case $\beta_1 = \beta_2$, and for $\eta = \beta_3/(\beta_1 + \beta_3) = \beta_3/(\beta_2 + \beta_3)$, the copula $C(u, v)$ becomes

$$C(u, v) = \begin{cases} uv^{1 - \eta} & \text{if } u < v \\ u^{1 - \eta}v & \text{if } u \geq v. \end{cases}$$

36.4 Some Special Cases

In this section we provide some special cases based on these two approaches. Different special cases have been considered in detail in the literature. In this section our main aim is to provide those special cases and mention relevant references associate with those models. We also provide some new bivariate models where more work can be done.

36.4.1 Model 1

Marshall-Olkin Bivariate Exponential Distribution

Marshall-Olkin bivariate exponential (MOBE) distribution seems to be the most popular bivariate distribution with a singular component, and it was originally introduced by Marshall and Olkin [44]. In this case the survival function of the base line distribution, namely, $S_B(t, \theta) = e^{-t}$, for $t > 0$, and zero, otherwise. Hence, U_1 , U_2 , and U_3 as defined in Sect. 36.3.1 follow $\text{Exp}(\alpha_1)$, $\text{Exp}(\alpha_2)$ and $\text{Exp}(\alpha_3)$, respectively. Hence, the joint PDF of X and Y becomes

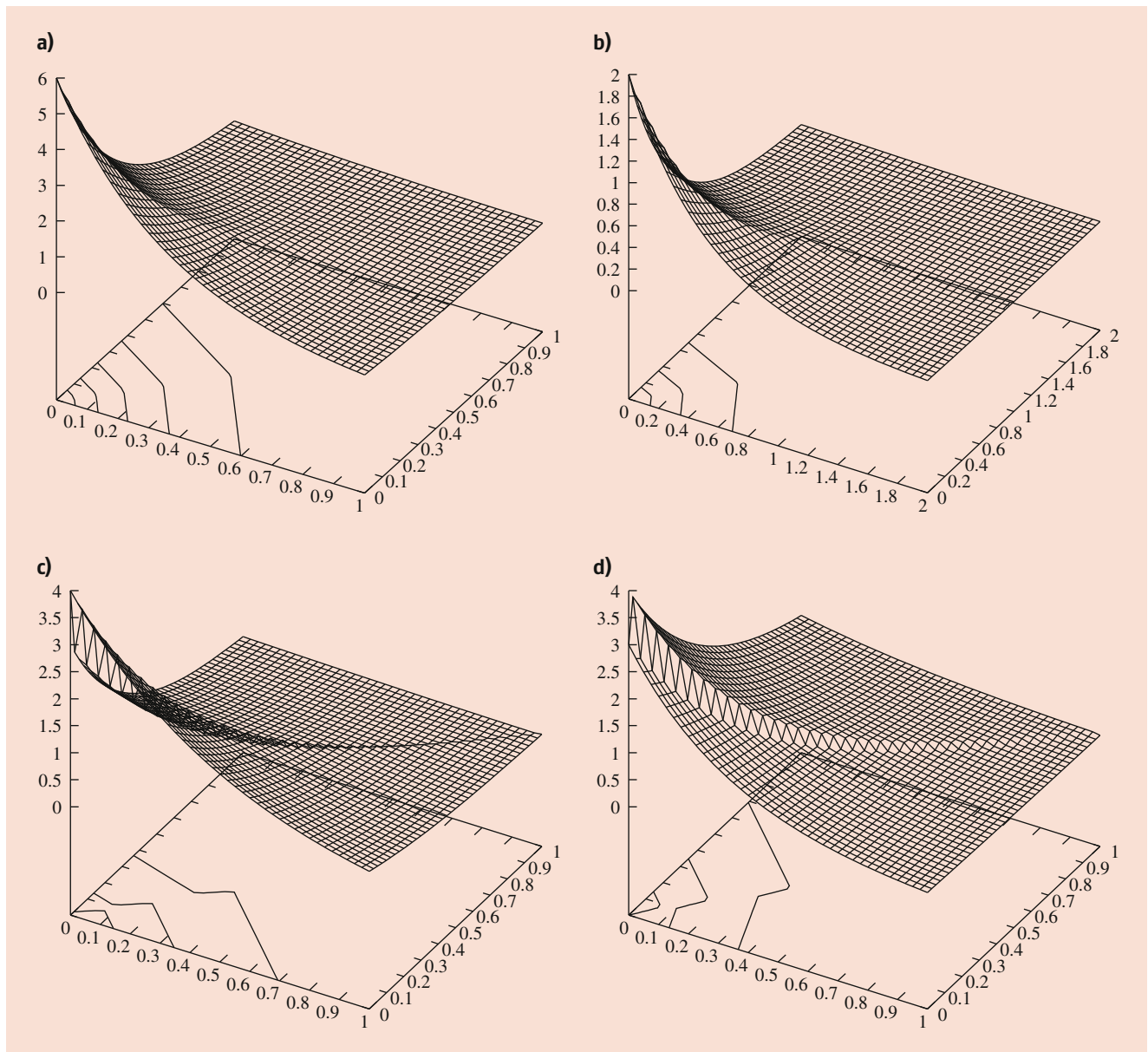


Fig. 36.1 PDF plots of $MOBE(\alpha_1, \alpha_2, \alpha_3)$ distribution for different $(\alpha_1, \alpha_2, \alpha_3)$ values: (a) (2.0,2.0,1.0) (b) (1.0,1.0,1.0) (c) (1.0,2.0,1.0) (d) (2.0,1.0,1.0)

$$f_{X,Y}(x, y) = \begin{cases} \alpha_1(\alpha_2 + \alpha_3)e^{-\alpha_1 x}e^{-(\alpha_2 + \alpha_3)y} & \text{if } x < y \\ \alpha_2(\alpha_1 + \alpha_3)e^{-(\alpha_1 + \alpha_3)x}e^{-\alpha_2 y} & \text{if } x > y \\ \alpha_3 e^{-(\alpha_1 + \alpha_2 + \alpha_3)x} & \text{if } x = y, \end{cases}$$

and it will be denoted by $MOBE(\alpha_1, \alpha_2, \alpha_3)$. The absolute continuous part of the PDF of $MOBE(\alpha_1, \alpha_2, \alpha_3)$ for different values of α_1, α_2 , and α_3 is provided in Fig. 36.1. It is clear from the figures that for all parameter values the maximum occurs at $(0, 0)$.

Note that the marginals of the MOBE distribution are exponential distributions. In this case X and Y follow $Exp(\alpha_1 + \alpha_3)$ and $Exp(\alpha_2 + \alpha_3)$, respectively. This model is also known

as the shock model since it has been introduced as modeling shocks to a parallel system and it has an interesting connection with the homogeneous Poisson process. Interested readers are referred to the original paper of Marshall and Olkin [44] in this respect. An extensive amount of work has been done dealing with different aspects of the MOBE model. Arnold [2] discussed the existence of the maximum likelihood estimators of the unknown parameters. Bemis et al. [11] and Bhattacharyya and Johnson [12] discussed different properties of the MOBE distribution. Pena and Gupta [54] developed the Bayesian inference of the unknown parameters based on a very flexible beta-gamma priors, and

Karlis [26] provided a very efficient EM algorithm to compute the maximum likelihood estimators of the unknown parameters.

Marshall-Olkin Bivariate Weibull Distribution

It can be seen that the MOBE has exponential marginals, and due to this reason it has some serious limitations. For example, if the data indicate that the marginals have unimodal PDFs, then clearly MOBE may not be used. Due to this reason, in the same paper Marshall and Olkin [44] introduced the Marshall Olkin bivariate Weibull (MOBW) model by replacing the exponential distribution with the Weibull distribution. In this case the base line survival function is a Weibull distribution, and it is taken as $S_B(t, \theta) = e^{-t^\theta}$, for $t > 0$, and zero otherwise. Hence, the base line distribution is a Weibull distribution with the shape parameter θ and scale parameter 1. Using the same notations as in Sect. 36.3.1, it can be easily seen that U_1 , U_2 , and U_3 follow Weibull distribution with the same shape parameter θ and having scale parameter α_1 , α_2 , and α_3 , respectively. Hence, the joint PDF of X and Y becomes

$$f_{X,Y}(x, y) = \begin{cases} \theta^2 \alpha_1 (\alpha_2 + \alpha_3) x^{\theta-1} y^{\theta-1} e^{-\alpha_1 x^\theta} e^{-(\alpha_2 + \alpha_3) y^\theta} & \text{if } x < y \\ \theta^2 \alpha_2 (\alpha_1 + \alpha_3) x^{\theta-1} y^{\theta-1} e^{-(\alpha_1 + \alpha_3) x^\theta} e^{-\alpha_2 y^\theta} & \text{if } x > y \\ \theta \alpha_3 x^{\theta-1} e^{-(\alpha_1 + \alpha_2 + \alpha_3) x^\theta} & \text{if } x = y, \end{cases} \quad (36.17)$$

and it will be denoted by $\text{MOBW}(\alpha_1, \alpha_2, \alpha_3, \theta)$. The absolute continuous part of the PDF of $\text{MOBW}(\alpha_1, \alpha_2, \alpha_3, \theta)$ for different values of α_1 , α_2 , α_3 , and θ is provided in Fig. 36.2. It is clear from the figures that for all parameter values the maximum occurs at $(0, 0)$ if the common shape parameter $0 < \theta \leq 1$; otherwise it is always unimodal.

Note that the marginals of the MOBW distributions are Weibull distributions with the same shape parameter, namely, X follows $\text{WE}(\theta, \alpha_1 + \alpha_3)$ and Y follows $\text{WE}(\theta, \alpha_2 + \alpha_3)$. This model has several real-life applications, and it has an interesting connection with the renewal process. An extensive amount of work has been done mainly related to the estimation of the unknown parameters both for classical and Bayesian methods. It may be noted that the maximum likelihood estimators of the unknown parameters cannot be obtained in closed form. It needs solving a four-dimensional optimization problem. Kundu and Dey [30] developed a very efficient EM algorithm to compute the maximum likelihood estimators of the unknown parameters which needs solving only one one-dimensional optimization problem. Kundu and Gupta [36] developed a very efficient Bayesian inference of the unknown based on a very flexible priors. Different methods have been evolved for analyzing censored data also. See, for example, Lu [41, 42]. Recently, Feizjavadian and Hashemi [18] and Shen and Xu [61] used MOBW distribution for analyzing dependent competing risks data. They have developed

very efficient EM algorithm to compute the known parameters of the model. It will be interesting to develop Bayesian inference of the unknown parameters in this case also.

Weighted Marshall-Olkin Bivariate Exponential Distribution

Jamalizadeh and Kundu [24] introduced the weighted Marshall-Olkin bivariate exponential distribution as an alternative to the MOBW distribution. It is also a bivariate singular distribution, and it can be a very flexible distribution similar to the MOBW distribution. It may be recalled that a random variable X is said to have a weighted exponential (WEE) distribution with parameters $\alpha > 0$ and $\lambda > 0$, if the PDF of X is of the form:

$$f_{WEE}(x; \alpha, \lambda) = \frac{\alpha + 1}{\alpha} \lambda e^{-\lambda x} (1 - e^{-\lambda x}); \quad x > 0,$$

and 0, otherwise. The WEE distribution was originally introduced by Gupta and Kundu [22] as an alternative to the two-parameter Weibull, gamma, or generalized exponential distributions. The PDFs and the hazard functions of the WEE distribution can take a variety of shapes similar to the Weibull, gamma, or generalized exponential distributions. The weighted Marshall-Olkin bivariate exponential (BWEE) distribution introduced by Jamalizadeh and Kundu [24] has the following PDF:

$$f_{X,Y}(x, y) = \begin{cases} \frac{\alpha + \lambda}{\alpha} \lambda_1 e^{-\lambda_1 x} (\lambda_2 + \lambda_3) e^{-(\lambda_2 + \lambda_3) y} (1 - e^{-x\alpha}) & \text{if } x < y \\ \frac{\alpha + \lambda}{\alpha} (\lambda_1 + \lambda_3) e^{-(\lambda_1 + \lambda_3) x} \lambda_2 e^{-\lambda_2 y} (1 - e^{-y\alpha}) & \text{if } x > y \\ \frac{\alpha + \lambda}{\alpha} \lambda_3 e^{-\lambda x} (1 - e^{-x\alpha}) & \text{if } x = y, \end{cases} \quad (36.18)$$

for $\lambda = \lambda_1 + \lambda_2 + \lambda_3$, and it will be denoted by $\text{BWEE}(\lambda_1, \lambda_2, \lambda_3, \alpha)$. The absolute continuous part of the PDF of $\text{BWEE}(\lambda_1, \lambda_2, \lambda_3, \alpha)$ for different values of λ_1 , λ_2 , λ_3 , and α is provided in Fig. 36.3. It is clear from the figures that for all parameter values the joint PDF is unimodal.

The marginals of the BWEE distribution are WEE distributions. Jamalizadeh and Kundu [24] established different properties of a BWEE distribution. The maximum likelihood estimators of the unknown parameters cannot be obtained in explicit forms. Efficient EM algorithm has been proposed by Jamalizadeh and Kundu [24] to compute the maximum likelihood estimators of the unknown parameters.

Bivariate Kumaraswamy Distribution

Barreto-Souza and Lemonte [9] considered the bivariate Kumaraswamy (BVK) distribution whose marginals are Kumaraswamy distribution. Since the Kumaraswamy distribution has the support on $[0, 1]$, the BVK distribution has the support $[0, 1] \times [0, 1]$. It may be recalled that a random variable X is said to have a Kumaraswamy distribution with parameters $\alpha > 0$ and $\beta > 0$, if it has the following CDF and PDF, respectively

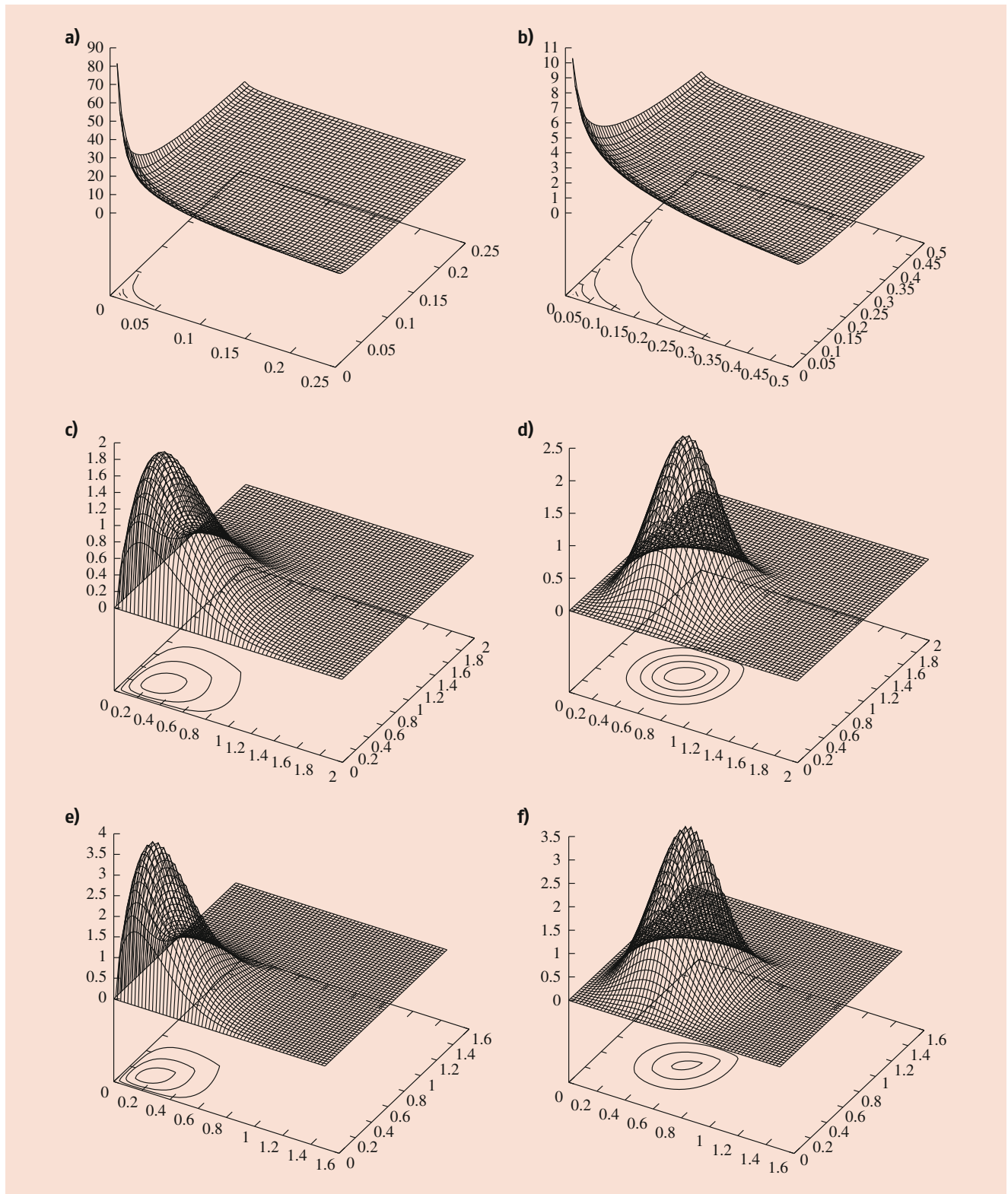


Fig. 36.2 PDF plots of $\text{MOBW}(\alpha_1, \alpha_2, \alpha_3, \theta)$ distribution for different $(\alpha_1, \alpha_2, \alpha_3, \theta)$ values: (a) (1.0,1.0,1.0,0.5) (b) (1.0,1.0,1.0,0.75) (c) (2.0,2.0,1.0,1.5) (d) (2.0,2.0,1.0,3.0) (e) (3.0,3.0,3.0,1.5) (f) (3.0,3.0,3.0,3.0)

$F_K(x; \alpha, \beta) = 1 - (1 - x^\beta)^\alpha$ and $f_K(x; \alpha, \beta) = \alpha\beta x^{\beta-1}(1 - x^\beta)^{\alpha-1}$, for $x > 0$. Hence, a BVK distribution with parameters $\alpha_1, \alpha_2, \alpha_3,$ and β has the following PDF:

$$f_{X,Y}(x, y) = \begin{cases} \alpha_1(\alpha_2 + \alpha_3)\beta^2 x^{\beta-1}(1 - x^\beta)^{\alpha_1-1} y^{\beta-1}(1 - y^\beta)^{\alpha_2+\alpha_3-1} & \text{if } x < y \\ (\alpha_1 + \alpha_3)\alpha_2\beta^2 x^{\beta-1}(1 - x^\beta)^{\alpha_1+\alpha_3-1} y^{\beta-1}(1 - y^\beta)^{\alpha_2-1} & \text{if } x > y \\ \theta\alpha_3\beta x^{\beta-1}(1 - x^\beta)^{\alpha_1+\alpha_2+\alpha_3-1} & \text{if } x = y, \end{cases} \quad (36.19)$$

and it will be denoted by $BVK(\alpha_1, \alpha_2, \alpha_3, \beta)$.

The PDF of the absolute continuous part of $BVK(\alpha_1, \alpha_2, \alpha_3, \beta)$ distribution for different $\alpha_1, \alpha_2, \alpha_3, \beta$ are provided in

Fig. 36.4. It is clear that it has bounded support on $[0,1] \times [0,1]$, and it can take a variety of shapes depending on the parameter values.

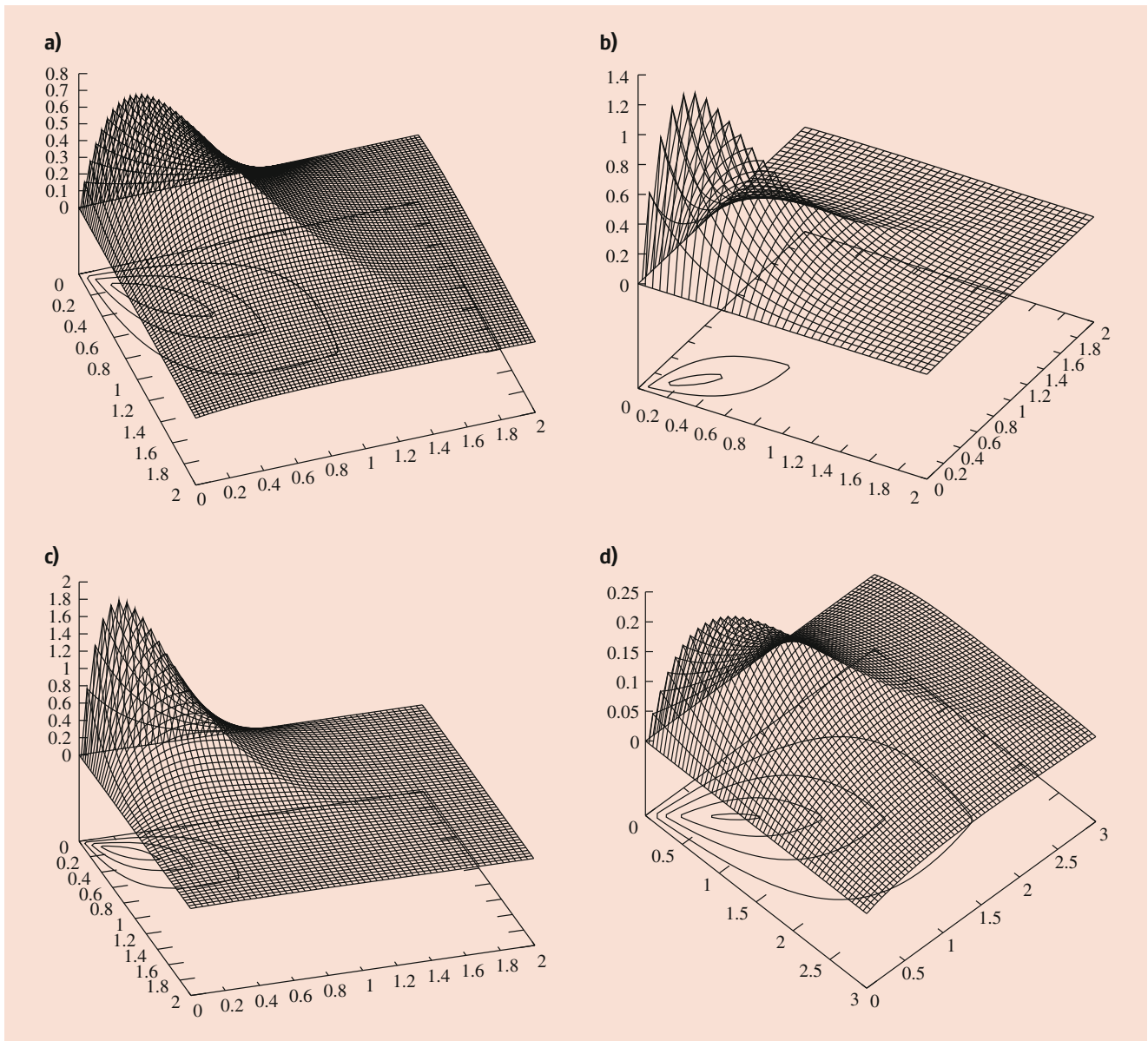


Fig. 36.3 PDF plots of $BWEE(\lambda_1, \lambda_2, \lambda_3, \alpha)$ distribution for different $(\lambda_1, \lambda_2, \lambda_3, \alpha)$ values: (a) (1.0,1.0,1.0,0.5) (b) (1.0,1.0,2.0,1.0) (c) (1.5,1.5,1.5,1.5) (d) (0.5,0.5,0.5,0.5)

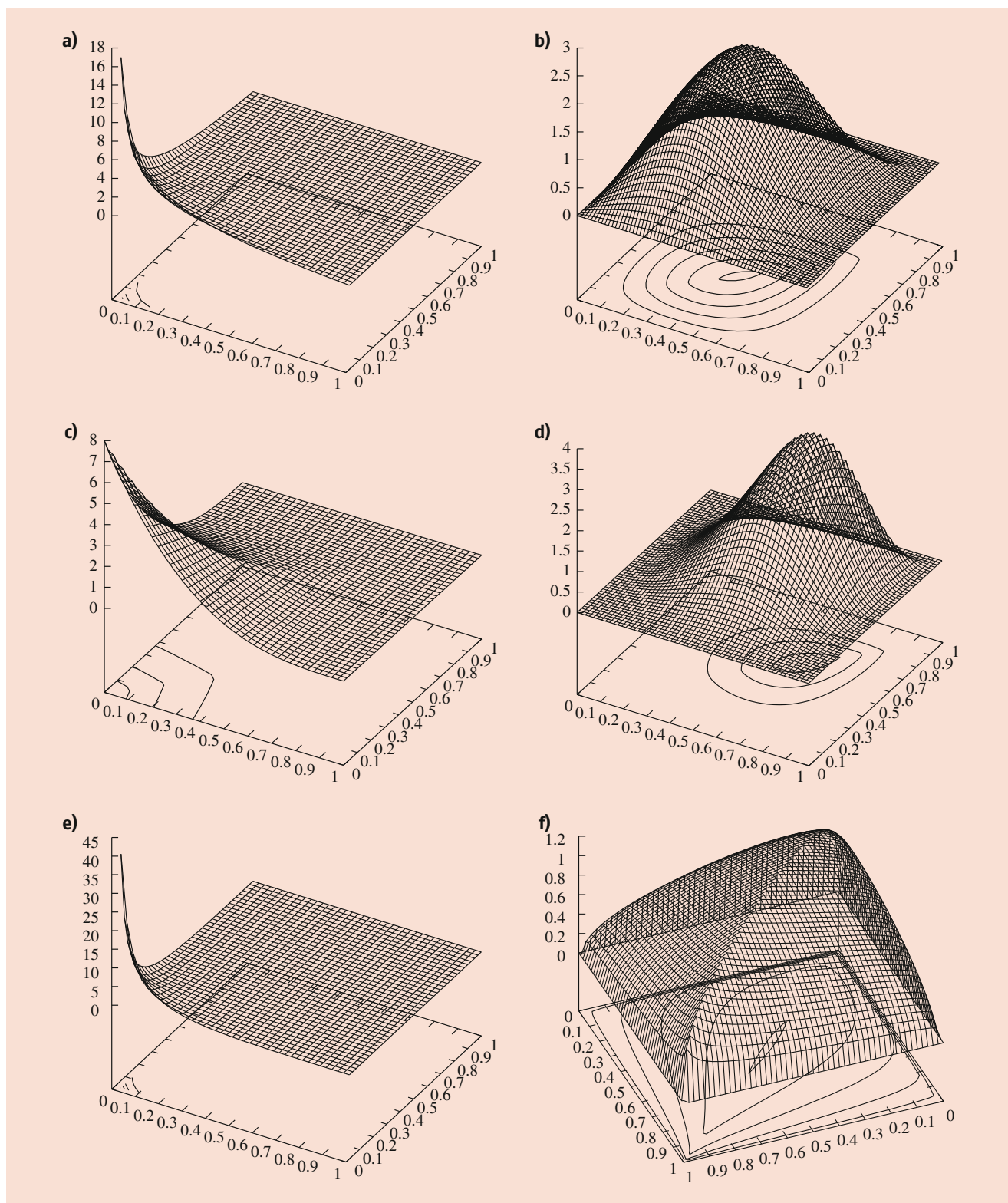


Fig. 36.4 PDF plots of $BVK(\alpha_1, \alpha_2, \alpha_3, \beta)$ distribution for different $(\alpha_1, \alpha_2, \alpha_3, \beta)$ values: (a) (1.0,1.0,1.0,0.5) (b) (2.0,2.0,2.0,2.0) (c) (2.0,2.0,2.0,1.0) (d) (2.0,2.0,2.0,3.0) (e) (2.0,2.0,2.0,0.5) (f) (0.75,0.75,0.75,1.1)

Barreto-Souza and Lemonte [9] developed several properties of the BVK distribution and also provided a very efficient EM algorithm to compute the maximum likelihood estimators of the unknown parameters. They have used this model to analyze a bivariate singular data set with bounded support.

36.4.2 Model 2

Bivariate Proportional Reversed Hazard Distribution

Kundu and Gupta [32] introduced the bivariate generalized exponential (BVGE) distribution as a stress model and as a

maintenance model. The idea is very similar to the MOBE or the MOBW distribution, but in this case the authors considered the maximization approach than the minimization method. It is assumed that the base line distribution function is an exponential distribution with the scale parameter λ , i.e., $F_B(t; \lambda) = (1 - e^{-t\lambda})$, for $t > 0$ and zero, otherwise. Let us assume that V_1 , V_2 , and V_3 have the CDFs $(1 - e^{-t\lambda})^{\beta_1}$, $(1 - e^{-t\lambda})^{\beta_2}$, and $(1 - e^{-t\lambda})^{\beta_3}$, and they are independently distributed. Here $\beta_1 > 0$, $\beta_2 > 0$, and $\beta_3 > 0$. It is clear that V_1 , V_2 , and V_3 have $GE(\beta_1, \lambda)$, $GE(\beta_2, \lambda)$, and $GE(\beta_3, \lambda)$, respectively. Hence, the joint PDF of (X, Y) in this case becomes

$$f_{X,Y}(x, y) = \begin{cases} (\beta_1 + \beta_3)\beta_2(1 - e^{-\lambda x})^{\beta_1 + \beta_3 - 1}(1 - e^{-\lambda y})^{\beta_2 - 1}e^{-\lambda(x+y)} & \text{if } 0 < x < y < \infty \\ (\beta_2 + \beta_3)\beta_1(1 - e^{-\lambda x})^{\beta_1 - 1}(1 - e^{-\lambda y})^{\beta_2 + \beta_3 - 1}e^{-\lambda(x+y)} & \text{if } 0 < y < x < \infty \\ \beta_3(1 - e^{-\lambda x})^{\beta_1 + \beta_2 + \beta_3 - 1}e^{-\lambda x} & \text{if } 0 < y = x < \infty, \end{cases} \quad (36.20)$$

and it will be denoted by $BVGE(\beta_1, \beta_2, \beta_3, \lambda)$. The absolute continuous part of the PDF of $BVGE(\beta_1, \beta_2, \beta_3, \lambda)$ for different values of β_1 , β_2 , β_3 , and λ is provided in Fig. 36.5. It is clear that the joint PDF of a BVGE is very similar to the joint PDF of a MOBW distribution for different parameter values. It is clear that if $0 < \beta_1 + \beta_3 < 1$ and $0 < \beta_1 + \beta_2 < 1$, then the maximum occurs at $(0, 0)$; otherwise it is unimodal.

It is observed that the BVGE distribution is also quite flexible like the BVWE distribution, and the marginals of the BVGE distribution follow generalized exponential distributions. Because of the presence of the four parameters, the BVGE distribution can be used quite effectively in analyzing various bivariate data sets. Kundu and Gupta [32] provided a very effective EM algorithm in computing the maximum likelihood estimators of the unknown parameters. It will be interesting to see how the EM algorithm can be modified for analyzing censored data also. No work has been done in developing Bayesian inference of the unknown parameters. It may be mentioned that as MOBW has been used for analyzing dependent competing risks data, BVGE distribution may be used for analyzing dependent complementary risks data; see, for example, Mondal and Kundu [47] in this respect.

Kundu and Gupta [34] extended the BVGE distribution to a more general class of bivariate proportional reversed hazard (BVPRH) distribution. In that paper the authors introduced three other classes of bivariate distributions, namely, (a) bivariate exponentiated Weibull (BVEW) distribution, (b) bivariate exponentiated Rayleigh (BVER) distribution, and (c) bivariate generalized linear failure rate (BVGLF) distribution. The BVEW distribution has been obtained by taking the base line distribution as $F_B(t; \alpha, \lambda) = (1 - e^{-\lambda t^\alpha})$, i.e., a

Weibull distribution with the scale parameter λ and the shape parameter α . The BVER distribution can be obtained by taking the base line distribution as a Rayleigh distribution, i.e., $F_B(t; \lambda) = (1 - e^{-\lambda t^2})$. Similarly, the BVGLF distribution can be obtained by taking $F_B(t; \lambda, \theta) = (1 - e^{-(\lambda t + \theta \lambda t^2)})$. Sarhan et al. [60] provided the detailed analysis of the BVGLF distribution. They obtained various properties and developed classical inference of the unknown parameters. It will be interesting to develop Bayesian inferences in all the above cases.

36.5 Classical Inference

In this section we present the classical inferences of the unknown parameters for both the classes of models. It may be mentioned that Arnold [2] first considered the MLEs of the unknown parameters for the Marshall-Olkin bivariate and multivariate normal distributions. Karlis [26] proposed the EM algorithm to compute the MLEs of the unknown parameters of the MOBE distribution. Kundu and Dey [30] extended the result of Karlis [26] to the case of MOBW distribution. In a subsequent paper, Kundu and Gupta [32] provided an EM algorithm to compute the MLEs of the unknown parameters for the modified Sarhan-Balakrishnan singular bivariate distribution. In this section we provide a general EM algorithm which can be used to compute the MLEs of the unknown parameters of bivariate distributions with singular components which can be obtained either by minimization or maximization approach.

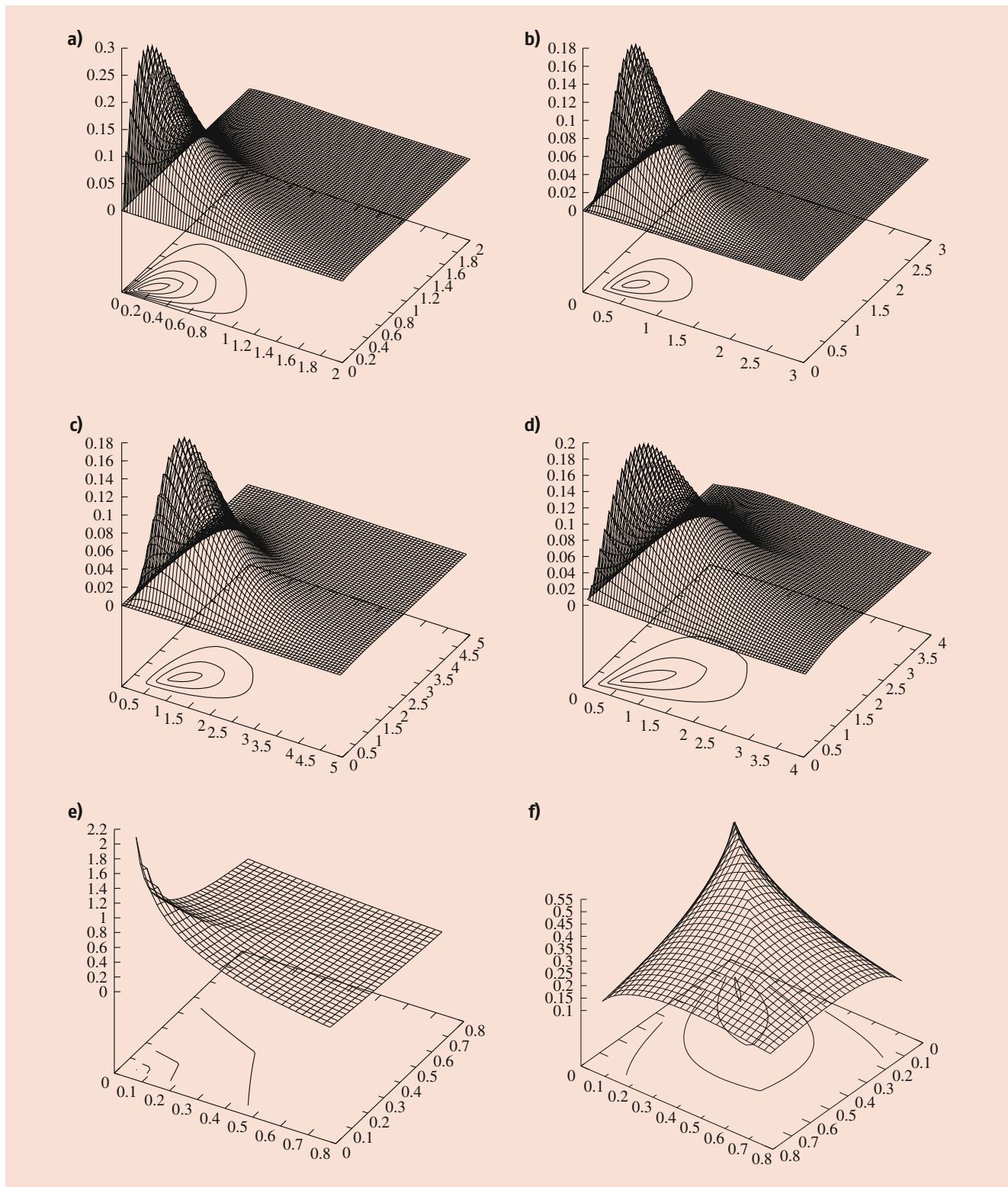


Fig. 36.5 PDF plots of $MOBE(\alpha_1, \alpha_2, \alpha_3, \theta)$ distribution for different $(\alpha_1, \alpha_2, \alpha_3, \theta)$ values: (a) (1.0,1.0,1.0,0.5) (b) (1.0,1.0,1.0,0.75) (c) (2.0,2.0,1.0,1.5) (d) (2.0,2.0,1.0,3.0) (e) (3.0,3.0,3.0,1.5) (f) (3.0,3.0,3.0,3.0)

36.5.1 EM Algorithm: Model 1

In this section we provide the EM algorithm to compute the MLEs of the unknown parameters when the data are coming from the joint PDF (36.11). It is assumed that we have the following data:

$$\mathcal{D} = \{(x_1, y_1), \dots, (x_n, y_n)\}.$$

It is assumed that $\alpha_1, \alpha_2, \alpha_3$, and θ are unknown parameters and the form of $S_B(t; \cdot)$ is known. It may be mentioned that θ

may be a vector valued also, but in this case we assume it to be a scalar for simplicity. Our method can be easily modified for the vector valued θ also. We further use the following notations:

$$I_0 = \{i : x_i = y_i\}, \quad I_1 = \{i : x_i < y_i\},$$

$$I_2 = \{i : x_i > y_i\}, \quad I = I_0 \cup I_1 \cup I_2,$$

$$|I_0| = n_0, \quad |I_1| = n_1, \quad |I_2| = n_2,$$

here $|I_j|$ for $j = 0, 1, 2$ denotes the number of elements in the set I_j . The log-likelihood function can be written as

$$\begin{aligned} l(\alpha_1, \alpha_2, \alpha_3, \theta) &= n_0 \ln \alpha_3 + n_1 \ln \alpha_1 + n_1 \ln(\alpha_2 + \alpha_3) + n_2 \ln \alpha_2 + n_2 \ln(\alpha_1 + \alpha_3) \\ &+ \sum_{i \in I} \ln f_B(x_i, \theta) + \sum_{i \in I_1 \cup I_2} \ln f_B(y_i, \theta) + (\alpha_1 + \alpha_2 + \alpha_3 - 1) \sum_{i \in I_0} \ln S_B(x_i, \theta) \\ &+ (\alpha_1 - 1) \sum_{i \in I_1} \ln S_B(x_i, \theta) + (\alpha_2 + \alpha_3 - 1) \sum_{i \in I_1} \ln S_B(y_i, \theta) \\ &+ (\alpha_1 + \alpha_3 - 1) \sum_{i \in I_1} \ln S_B(x_i, \theta) + (\alpha_2 - 1) \sum_{i \in I_1} \ln S_B(y_i, \theta). \end{aligned} \quad (36.21)$$

Hence, the MLEs of the unknown parameters can be obtained by maximizing (36.21) with respect to the unknown parameters. It is known that even in special cases, i.e., in case of exponential and Weibull distribution (see, e.g., Bemis et al. [11] and Kundu and Dey [30]), the MLEs do not exist if $n_1 n_2 n_3 = 0$. If $n_1 n_2 n_3 > 0$, then the MLEs exist, but they cannot be obtained in explicit forms. The MLEs have to be obtained by solving nonlinear equations, and it becomes a nontrivial problem. Note that in case of exponential, one needs to solve a three-dimensional optimization problem, and in case of Weibull, it is a four-dimensional optimization problem. Due to this reason several approximations and alternative estimators have been proposed in the literature; see, for example, Arnold [2] and Proschan and Sullo [55] in this respect.

We propose an EM algorithm which can be used to compute the MLEs of the unknown parameters, and it is an extension of the EM algorithm originally proposed by Karlis [26] for the exponential distribution. The basic idea of the proposed EM algorithm is quite simple. It may be observed that if instead of (X, Y) , (U_1, U_2, U_3) are known, then the MLEs of the unknown parameters can be obtained quite conveniently. Suppose we have the following complete data:

$$\mathcal{D}^c = \{(u_{11}, u_{12}, u_{13}), \dots, (u_{n1}, u_{n2}, u_{n3})\}, \quad (36.22)$$

then the log-likelihood function of the complete data can be written as

$$\begin{aligned} l_c(\alpha_1, \alpha_2, \alpha_3, \theta) &= n \ln \alpha_1 + \alpha_1 \sum_{i=1}^n \ln S_B(u_{i1}, \theta) + \sum_{i=1}^n (\ln f_B(u_{i1}, \theta) + \ln S_B(u_{i1}, \theta)) \\ &+ n \ln \alpha_2 + \alpha_2 \sum_{i=1}^n \ln S_B(u_{i2}, \theta) + \sum_{i=1}^n (\ln f_B(u_{i2}, \theta) + \ln S_B(u_{i2}, \theta)) \\ &+ n \ln \alpha_3 + \alpha_3 \sum_{i=1}^n \ln S_B(u_{i3}, \theta) + \sum_{i=1}^n (\ln f_B(u_{i3}, \theta) + \ln S_B(u_{i3}, \theta)). \end{aligned} \quad (36.23)$$

Hence, for a fixed θ , the MLEs of α_1 , α_2 , and α_3 can be obtained as

$$\widehat{\alpha}_{1c}(\theta) = -\frac{n}{\sum_{i=1}^n \ln S_B(u_{i1}, \theta)}, \quad \widehat{\alpha}_{2c}(\theta) = -\frac{n}{\sum_{i=1}^n \ln S_B(u_{i2}, \theta)}, \quad \widehat{\alpha}_{3c}(\theta) = -\frac{n}{\sum_{i=1}^n \ln S_B(u_{i3}, \theta)}. \quad (36.24)$$

Once, $\widehat{\alpha}_{1c}(\theta)$, $\widehat{\alpha}_{2c}(\theta)$, and $\widehat{\alpha}_{3c}(\theta)$ are obtained, the MLE of θ can be obtained by maximizing the profile log-likelihood function $l_c(\widehat{\alpha}_{1c}, \widehat{\alpha}_{2c}, \widehat{\alpha}_{3c}, \theta)$ with respect to θ . Therefore, instead of solving a four-dimensional optimization problem, one needs to solve only a one-dimensional optimization problem in this case. Due to this reason we treat this problem

as a missing value problem. The Table 36.1 will be useful in identifying the missing U_i s in different cases and the associated probabilities.

We need the following derivations for developing the EM algorithm for $i, j = 1, 2, 3$ and $i \neq j$.

$$E(U_j | U_j > u) = \frac{1}{[S_B(u, \theta)]^{\alpha_j}} \int_u^\infty \alpha_j t f_B(t; \theta) [S_B(t; \theta)]^{\alpha_j - 1} dt \quad (36.25)$$

$$E(U_j | \min\{U_i, U_j\} = u) = \frac{\alpha_j}{\alpha_i + \alpha_j} u + \frac{\alpha_i}{\alpha_i + \alpha_j} \times \frac{1}{[S_B(u, \theta)]^{\alpha_j}} \int_u^\infty \alpha_j t f_B(t; \theta) [S_B(t; \theta)]^{\alpha_j - 1} dt. \quad (36.26)$$

Using (36.25) and (36.26), we estimate the missing U_i s by its expected value. The expectation either can be performed by a direct numerical integration or by Monte Carlo simulations. Therefore, the following EM algorithm can be used to compute the MLEs of the unknown parameters in this case. Start with an initial guess of α_1 , α_2 , α_3 , and θ as

$\alpha_1^{(0)}$, $\alpha_2^{(0)}$, $\alpha_3^{(0)}$, and $\theta^{(0)}$, respectively. We provide the explicit method how the $(k+1)$ -th iterate can be obtained from the k -th iterate. Suppose at the k -th iterate the values of α_1 , α_2 , α_3 , and θ are $\alpha_1^{(k)}$, $\alpha_2^{(k)}$, $\alpha_3^{(k)}$, and $\theta^{(k)}$, respectively. Then if a data point $(x_i, x_i) \in I_0$, clearly, $u_{i3} = x_i$, and the missing u_{i1} and u_{i2} can be obtained as

$$u_{i1}^{(k)} = \frac{1}{[S_B(x_i, \theta^{(k)})]^{\alpha_1^{(k)}}} \int_{x_i}^\infty \alpha_1^{(k)} t f_B(t; \theta^{(k)}) [S_B(t; \theta^{(k)})]^{\alpha_1^{(k)} - 1} dt, \quad \text{and}$$

$$u_{i2}^{(k)} = \frac{1}{[S_B(x_i, \theta^{(k)})]^{\alpha_2^{(k)}}} \int_{x_i}^\infty \alpha_2^{(k)} t f_B(t; \theta^{(k)}) [S_B(t; \theta^{(k)})]^{\alpha_2^{(k)} - 1} dt.$$

Similarly, if $(x_i, y_i) \in I_1$, then $u_{i1} = x_i$, and the missing u_{i2} and u_{i3} can be obtained as

$$u_{i2}^{(k)} = \frac{y_i \alpha_2^{(k)}}{\alpha_2^{(k)} + \alpha_3^{(k)}} + \frac{\alpha_3^{(k)}}{\alpha_2^{(k)} + \alpha_3^{(k)}} \times \frac{1}{[S_B(x_i, \theta^{(k)})]^{\alpha_2^{(k)}}} \int_{y_i}^\infty \alpha_2^{(k)} t f_B(t; \theta^{(k)}) [S_B(t; \theta^{(k)})]^{\alpha_2^{(k)} - 1} dt,$$

$$u_{i3}^{(k)} = \frac{y_i \alpha_3^{(k)}}{\alpha_2^{(k)} + \alpha_3^{(k)}} + \frac{\alpha_2^{(k)}}{\alpha_2^{(k)} + \alpha_3^{(k)}} \times \frac{1}{[S_B(x_i, \theta^{(k)})]^{\alpha_3^{(k)}}} \int_{y_i}^\infty \alpha_3^{(k)} t f_B(t; \theta^{(k)}) [S_B(t; \theta^{(k)})]^{\alpha_3^{(k)} - 1} dt.$$

If $(x_i, y_i) \in I_2$, then $u_{i2} = y_i$, and the missing u_{i1} and u_{i3} can be obtained as

$$u_{i1}^{(k)} = \frac{x_i \alpha_1^{(k)}}{\alpha_1^{(k)} + \alpha_3^{(k)}} + \frac{\alpha_3^{(k)}}{\alpha_1^{(k)} + \alpha_3^{(k)}} \times \frac{1}{[S_B(y_i, \theta^{(k)})]^{\alpha_1^{(k)}}} \int_{x_i}^{\infty} \alpha_1^{(k)} t f_B(t; \theta^{(k)}) [S_B(t; \theta^{(k)})]^{\alpha_2^{(k)} - 1} dt,$$

$$u_{i3}^{(k)} = \frac{x_i \alpha_3^{(k)}}{\alpha_1^{(k)} + \alpha_3^{(k)}} + \frac{\alpha_1^{(k)}}{\alpha_1^{(k)} + \alpha_3^{(k)}} \times \frac{1}{[S_B(y_i, \theta^{(k)})]^{\alpha_3^{(k)}}} \int_{x_i}^{\infty} \alpha_3^{(k)} t f_B(t; \theta^{(k)}) [S_B(t; \theta^{(k)})]^{\alpha_2^{(k)} - 1} dt.$$

Therefore, we can obtain $\widehat{\alpha}_{1c}^{(k+1)}(\theta)$, $\widehat{\alpha}_{2c}^{(k+1)}(\theta)$, and $\widehat{\alpha}_{3c}^{(k+1)}(\theta)$ from the Eq. (36.24) by replacing (i) u_{i1} by $u_{i1}^{(k)}$ if $i \in I_0 \cup I_2$, (ii) u_{i2} by $u_{i2}^{(k)}$ if $i \in I_0 \cup I_1$, and (iii) u_{i3} by $u_{i3}^{(k)}$ if $i \in I_1 \cup I_2$. Obtain $\theta^{(k+1)}$ as

$$\theta^{(k+1)} = \arg \max_{\theta} l_c(\widehat{\alpha}_{1c}^{(k+1)}(\theta), \widehat{\alpha}_{2c}^{(k+1)}(\theta), \widehat{\alpha}_{3c}^{(k+1)}(\theta), \theta).$$

Here the function $l_c(\cdot, \cdot, \cdot, \cdot)$ is the log-likelihood function of the complete data set as defined by (36.23). Once we obtain $\theta^{(k+1)}$, $\alpha_1^{(k+1)}$, $\alpha_2^{(k+1)}$, and $\alpha_3^{(k+1)}$ can be obtained as

$$\alpha_1^{(k+1)} = \widehat{\alpha}_{1c}^{(k+1)}(\theta^{(k+1)}), \quad \alpha_2^{(k+1)} = \widehat{\alpha}_{2c}^{(k+1)}(\theta^{(k+1)}), \quad \alpha_3^{(k+1)} = \widehat{\alpha}_{3c}^{(k+1)}(\theta^{(k+1)}).$$

The process continues unless convergence takes place.

and I_2 as defined before. In this case the unknown parameters are β_1, β_2 , and β_3 and λ . In general λ can be vector valued also, but for simplicity it has been assumed to be a scalar valued. If it is assumed that the complete data are of the form

$$\mathcal{D}^c = \{(v_{11}, v_{12}, v_{13}), \dots, (v_{n1}, v_{n2}, v_{n3})\},$$

In this section we provide the EM algorithm when the data are coming from a bivariate distribution with the joint PDF (36.15). In this case we have assumed that the data are of the same form as in the previous case, and we have the sets I_0, I_1 ,

then the log-likelihood function based on the complete data can be written as

$$l_c(\beta_1, \beta_2, \beta_3, \lambda) = n \ln \beta_1 + \beta_1 \sum_{i=1}^n \ln F_B(v_{i1}, \lambda) + \sum_{i=1}^n (\ln f_B(v_{i1}, \theta) + \ln F_B(v_{i1}, \theta))$$

$$n \ln \beta_2 + \beta_2 \sum_{i=1}^n \ln F_B(v_{i2}, \theta) + \sum_{i=1}^n (\ln f_B(v_{i2}, \theta) + \ln F_B(v_{i2}, \theta))$$

$$n \ln \beta_3 + \beta_3 \sum_{i=1}^n \ln F_B(v_{i3}, \theta) + \sum_{i=1}^n (\ln f_B(v_{i3}, \theta) + \ln F_B(v_{i3}, \theta)). \tag{36.27}$$

Hence, for a fixed λ , the MLEs of β_1, β_2 , and β_3 can be obtained as

$$\widehat{\beta}_{1c}(\lambda) = -\frac{n}{\sum_{i=1}^n \ln F_B(v_{i1}, \lambda)}, \quad \widehat{\beta}_{2c}(\lambda) = -\frac{n}{\sum_{i=1}^n \ln F_B(v_{i2}, \lambda)}, \quad \widehat{\beta}_{3c}(\lambda) = -\frac{n}{\sum_{i=1}^n \ln F_B(v_{i3}, \lambda)}. \tag{36.28}$$

As before, once, $\widehat{\beta}_{1c}(\lambda)$, $\widehat{\beta}_{2c}(\lambda)$, and $\widehat{\beta}_{3c}(\lambda)$ are obtained, the MLE of λ can be obtained by maximizing the profile log-likelihood function $l_c(\widehat{\beta}_{1c}(\lambda), \widehat{\beta}_{2c}(\lambda), \widehat{\beta}_{3c}(\lambda), \lambda)$ with respect

to λ . We have a similar Table 36.2 as Table 36.1, identifying the missing V_i s in different cases. Similar to (36.25) and (36.26), we need the following expressions in this case for developing the EM algorithm for $i, j = 1, 2, 3$ and $i \neq j$.

$$E(V_j | V_j < u) = \frac{1}{[F_B(u, \lambda)]^{\beta_j}} \int_0^u \beta_j t f_B(t; \lambda) [F_B(t; \lambda)]^{\beta_j - 1} dt \quad (36.29)$$

$$E(V_j | \max\{V_i, V_j\} = u) = \frac{\beta_j}{\beta_i + \beta_j} u + \frac{\beta_i}{\beta_i + \beta_j} \frac{1}{[F_B(u, \lambda)]^{\beta_j}} \int_0^u \beta_j t f_B(t; \lambda) [F_B(t; \lambda)]^{\beta_j - 1} dt. \quad (36.30)$$

Now we provide the explicit method in this case how the $(k + 1)$ -th iterate can be obtained from the k -th iterate. Let us assume that at the k -th iterate the values of $\beta_1, \beta_2, \beta_3$, and λ are $\beta_1^{(k)}, \beta_2^{(k)}, \beta_3^{(k)}$, and $\lambda^{(k)}$, respectively. Then if a data point $(x_i, x_i) \in I_0$, clearly, $u_{i3} = x_i$, and the missing v_{i1} and v_{i2} can be obtained as

$$v_{i1}^{(k)} = \frac{1}{[F_B(x_i, \lambda^{(k)})]^{\beta_1^{(k)}}} \int_0^{x_i} \beta_1^{(k)} t f_B(t; \lambda^{(k)}) [F_B(t; \lambda^{(k)})]^{\beta_1^{(k)} - 1} dt,$$

and

$$v_{i2}^{(k)} = \frac{1}{[F_B(x_i, \lambda^{(k)})]^{\beta_2^{(k)}}} \int_0^{x_i} \beta_2^{(k)} t f_B(t; \lambda^{(k)}) [F_B(t; \lambda^{(k)})]^{\beta_2^{(k)} - 1} dt.$$

Similarly, if $(x_i, y_i) \in I_1$, then $v_{i2} = y_i$, and the missing v_{i1} and v_{i3} can be obtained as

$$v_{i1}^{(k)} = \frac{x_i \beta_1^{(k)}}{\beta_1^{(k)} + \beta_3^{(k)}} + \frac{\beta_3^{(k)}}{\beta_1^{(k)} + \beta_3^{(k)}} \times \frac{1}{[F_B(x_i, \lambda^{(k)})]^{\beta_1^{(k)}}} \int_0^{x_i} \beta_1^{(k)} t f_B(t; \lambda^{(k)}) [F_B(t; \lambda^{(k)})]^{\beta_1^{(k)} - 1} dt,$$

$$v_{i3}^{(k)} = \frac{x_i \beta_3^{(k)}}{\beta_1^{(k)} + \beta_3^{(k)}} + \frac{\beta_1^{(k)}}{\beta_1^{(k)} + \beta_3^{(k)}} \times \frac{1}{[F_B(x_i, \lambda^{(k)})]^{\beta_3^{(k)}}} \int_0^{x_i} \beta_3^{(k)} t f_B(t; \lambda^{(k)}) [F_B(t; \lambda^{(k)})]^{\beta_3^{(k)} - 1} dt.$$

If $(x_i, y_i) \in I_2$, then $v_{i1} = x_i$, and the missing v_{i2} and v_{i3} can be obtained as

$$v_{i2}^{(k)} = \frac{y_i \beta_2^{(k)}}{\beta_2^{(k)} + \beta_3^{(k)}} + \frac{\beta_3^{(k)}}{\beta_2^{(k)} + \beta_3^{(k)}} \times \frac{1}{[F_B(y_i, \lambda^{(k)})]^{\beta_2^{(k)}}} \int_0^{y_i} \beta_2^{(k)} t f_B(t; \lambda^{(k)}) [F_B(t; \lambda^{(k)})]^{\beta_2^{(k)} - 1} dt,$$

$$v_{i3}^{(k)} = \frac{y_i \beta_3^{(k)}}{\beta_2^{(k)} + \beta_3^{(k)}} + \frac{\beta_2^{(k)}}{\beta_2^{(k)} + \beta_3^{(k)}} \times \frac{1}{[F_B(y_i, \lambda^{(k)})]^{\beta_3^{(k)}}} \int_0^{y_i} \beta_3^{(k)} t f_B(t; \lambda^{(k)}) [F_B(t; \lambda^{(k)})]^{\beta_3^{(k)} - 1} dt.$$

In this case also similarly as before, we can obtain $\widehat{\beta}_{1c}^{(k+1)}(\lambda)$, $\widehat{\beta}_{2c}^{(k+1)}(\lambda)$, and $\widehat{\beta}_{3c}^{(k+1)}(\lambda)$ from the Eq. (36.28) by replacing (i) v_{i1} by $v_{i1}^{(k)}$ if $i \in I_0 \cup I_1$, (ii) v_{i2} by $v_{i2}^{(k)}$ if $i \in I_0 \cup I_2$, and (iii) v_{i3} by $v_{i3}^{(k)}$ if $i \in I_1 \cup I_2$. Obtain $\lambda^{(k+1)}$ as

$$\lambda^{(k+1)} = \arg \max l_c(\widehat{\beta}_{1c}^{(k+1)}(\lambda), \widehat{\beta}_{2c}^{(k+1)}(\lambda), \widehat{\beta}_{3c}^{(k+1)}(\lambda), \lambda).$$

Here the function $l_c(\cdot, \cdot, \cdot, \cdot)$ is the log-likelihood function of the complete data set as defined by (36.27). Once we obtain $\lambda^{(k+1)}, \beta_1^{(k+1)}, \beta_2^{(k+1)}$, and $\beta_3^{(k+1)}$ can be obtained as

$$\begin{aligned} \beta_1^{(k+1)} &= \widehat{\beta}_{1c}^{(k+1)}(\lambda^{(k+1)}), & \beta_2^{(k+1)} &= \widehat{\beta}_{2c}^{(k+1)}(\lambda^{(k+1)}), \\ \beta_3^{(k+1)} &= \widehat{\beta}_{3c}^{(k+1)}(\lambda^{(k+1)}). \end{aligned}$$

The process continues unless convergence takes place.

Table 36.1 Different cases and missing U_i s

Different cases	$X \& Y$	Set	Value of U_i 's	Missing U_i 's	Probability of Missing U_i 's
$U_3 < \min\{U_1, U_2\}$	$X = Y$	I_0	$U_3 = X = Y$	$U_1 \& U_2$	1
$U_1 < \min\{U_2, U_3\}$	$X < Y$	I_1	$X = U_1, Y = U_2$	U_3	$\frac{\alpha_2}{\alpha_2 + \alpha_3}$
$U_1 < \min\{U_2, U_3\}$	$X < Y$	I_1	$X = U_1, Y = U_3$	U_2	$\frac{\alpha_3}{\alpha_2 + \alpha_3}$
$U_2 < \min\{U_1, U_3\}$	$Y < X$	I_2	$X = U_1, Y = U_2$	U_3	$\frac{\alpha_1}{\alpha_1 + \alpha_3}$
$U_2 < \min\{U_1, U_3\}$	$Y < X$	I_2	$X = U_3, Y = U_2$	U_1	$\frac{\alpha_3}{\alpha_1 + \alpha_3}$

Table 36.2 Different cases and missing V_i s

Different cases	$X \& Y$	Set	Value of V_i s	Missing V_i s	Probability of Missing V_i s
$V_3 > \max\{V_1, V_2\}$	$X = Y$	I_0	$V_3 = X = Y$	$V_1 \& V_2$	1
$V_2 > \max\{V_1, V_3\}$	$X < Y$	I_1	$X = V_1, Y = V_2$	V_3	$\frac{\beta_1}{\beta_1 + \beta_3}$
$V_2 > \max\{V_1, V_3\}$	$X < Y$	I_1	$X = V_3, Y = V_2$	V_1	$\frac{\beta_3}{\beta_1 + \beta_3}$
$V_1 > \max\{V_2, V_3\}$	$Y < X$	I_2	$X = V_1, Y = V_2$	V_3	$\frac{\beta_2}{\beta_2 + \beta_3}$
$V_1 > \max\{V_2, V_3\}$	$Y < X$	I_2	$X = V_1, Y = V_3$	V_2	$\frac{\beta_3}{\beta_2 + \beta_3}$

In both the cases the associated Fisher information matrix can be easily obtained at the last step of the EM algorithm, and it can be used to construct confidence intervals of the unknown parameters.

36.6 Data Analysis

In this section we provide the analysis of one data set mainly to show how the proposed EM algorithms work in practice. It is observed that it might be possible to provide some physical justification also in certain cases under some simplified assumptions.

This data set is originally available in Meintanis [46], and it is presented in Table 36.3. This data set represents the soccer data where at least one goal is scored by the home team and at least one goal is scored directly from a penalty kick, foul kick, or any other direct kick by any team. All these direct goals are usually called as kick goals. Here X represents the time in minutes of the first kick goal scored by any team, and Y represents the time in minutes the first goal of any type scored by the home team. Clearly, in this case all the three possibilities are present, namely, $X < Y, X > Y,$ and $X = Y$.

Let us consider three random variables $U_1, U_2,$ and U_3 as follows:

U_1 = time in minutes that a first kick goal has been scored by the opponent

U_2 = time in minutes that a first non-kick goal has been scored by the home team

U_3 = the time in minutes that a first kick goal has been scored by the home team

Table 36.3 UEFA Champion League data

2005–2006	X	Y	2004–2005	X	Y
Lyon-Real Madrid	26	20	Internazionale-Bremen	34	34
Milan-Fenerbahce	63	18	Real Madrid-Roma	53	39
Chensea-Anderlecht	19	19	Man. United-Fenerbahce	54	7
Club Brugge-Juventus	66	85	Bayern-Ajax	51	28
Fenerbahce-PSV	40	40	Moscov-PSG	76	64
Internazionale-Rangers	49	49	Barcelona-Shakhtar	64	15
Panathinaikos-Bremen	8	8	Leverkusen-Roma	26	48
Ajax-Arsenal	69	71	Arsenal-Panathinaikos	16	16
Man. United-Benfica	39	39	Bayern-M. TelAviv	55	11
Juventus-Bayern	66	62	Bremen-Internazionale	49	49
Club Brugge-Rapid	25	9	Anderlecht-Valencia	24	24
Olympiacos-Lyon	41	3	Panathinaikos-PSV	44	30
Internazionale-Porto	16	75	Arsenal-Rosenborg	42	3
Schalke-PSV	18	18	Liverpool-Olympiacos	27	47
Barcelona-Bremen	22	14	M. TelAviv-Juventus	28	28
Milan-Schalke	42	42	Bremen-Panathinaikos	2	2
Rapid-Juventus	36	52			

In this case $X = \min\{U_1, U_3\}$ and $Y = \min\{U_2, U_3\}$. If it is assumed that $U_1, U_2,$ and U_3 are independently distributed, then (X, Y) can be obtained as Model 1. In this case we have used three different $S_B(t; \theta)$ and analyze the data set. The following $S_B(t; \theta)$ has been used in this case:

(a) $S_B(t; \theta) = e^{-t},$ (b) $S_B(t; \theta) = e^{-t^\beta},$ (c) $S_B(t; \theta) = e^{-t^\beta}.$

Note that $S_B(t; \theta)$ in three different cases are as follows: (a) exponential, (b) Rayleigh, and (c) Weibull, and the corresponding hazard functions are (a) constant, (b) increasing, and (c) increasing or decreasing. We have divided all the data points by 100, and it is not going to make any difference in the inference procedure. We compute the MLEs of the

Table 36.4 MLEs of the different parameters, 95% confidence intervals, and associated log-likelihood values

$S_B(t; \theta)$	α_1	α_2	α_3	θ	ll
e^{-t}	0.7226 (0.1844, 1.2608)	1.6352 (0.8877, 2.3826)	1.7676 (1.0378, 2.4975)	—	-22.757
e^{-t^2}	1.7523 (0.5687, 2.3212)	3.2157 (2.1033, 4.3281)	2.9556 (1.9967, 3.9145)	—	-18.342
e^{-t^θ}	1.2192 (0.2708, 2.1415)	2.8052 (2.0921, 3.5184)	2.6927 (1.5011, 3.8852)	1.6954 (1.3248, 2.0623)	-13.118

Table 36.5 MLEs of the different parameters, 95% confidence intervals, and associated log-likelihood values

$F_B(t; \lambda)$	β_1	β_2	β_3	λ	α	ll
$1 - e^{-\lambda t}$	1.4552 (0.6572, 2.2334)	0.4686 (0.1675, 0.7694)	1.1703 (0.6512, 1.6894)	3.8991 (2.8009, 4.4991)	—	-20.592
$1 - e^{-\lambda t^2}$	0.4921 (0.1704, 0.7343)	0.1659 (0.0648, 0.2687)	0.4101 (0.2466, 0.5748)	4.0263 (2.5141, 5.5382)	—	-22.737
$1 - e^{-\lambda t^\alpha}$	1.2071 (0.5612, 1.8534)	0.3387 (0.0888, 0.5893)	0.8434 (0.4187, 1.2695)	3.852 (2.7163, 4.9897)	1.1328 (0.7154, 1.5102)	-17.236

unknown parameters based on the proposed EM algorithm. We start the EM algorithm with some initial guesses and continue the process. We stop the EM algorithm when the ratio $|l(k+1) - l(k)|/|l(k)| < 10^{-8}$; here $l(k)$ denotes the value of the log-likelihood function at the k -th iterate. Based on the method proposed by Louis [40], it is possible to compute the observed Fisher information matrix at the final step of the EM algorithm. We have provided the MLEs of the unknown parameters, the associate 95% confidence intervals, and the corresponding log-likelihood values for different cases in Table 36.4.

For illustrative purposes, we have used Model 2 also to analyze this data set. We have used three different $F_B(t; \lambda)$, Namely, (i) $F_B(t; \lambda) = (1 - e^{-\lambda t})$, (ii) $F_B(t; \lambda) = (1 - e^{-\lambda t^2})$, and (iii) $F_B(t; \lambda_1, \lambda_2) = (1 - e^{-\lambda_1 t^{\lambda_2}})$. In this case (i) represents exponential, (ii) represents Rayleigh, and (iii) represents Weibull distributions. In this case also we have calculated the MLEs of the unknown parameters using EM algorithm as mentioned before, and we have used the same stopping rule as in the previous case here also. The MLEs of the unknown parameters, the associated 95% confidence intervals, and the corresponding log-likelihood values are reported in Table 36.5. In all the cases it is observed that the EM algorithm converges within 25 iterations. It seems EM algorithm works quite well for both the models.

36.7 Some Other Bivariate Distributions

In this section we provide few other bivariate distributions with singular components which are available in the literature, which cannot be obtained as special cases of Model

1 or Model 2. It can be seen that in the first two examples, although the construction remains the same, their final forms are different. Reasons will be clear soon. Efficient estimation procedures using EM algorithm can be developed in both the cases. Marshall and Olkin [45] proposed a method to introduce an extra parameter to an univariate distribution, and they have shown how it can be done in case of an exponential or Weibull distribution. They have indicated how it can be done for the bivariate cases also. Kundu and Gupta [36] developed bivariate distributions with singular components based on the idea of Marshall and Olkin [45]. Different properties and efficient estimation procedures have been developed.

36.7.1 Sarhan-Balakrishnan Bivariate Distribution

Sarhan and Balakrishnan [58] proposed this distribution based on minimization approach, but they have not taken the base line distribution as the proportional hazard class. Sarhan and Balakrishnan [58] assumed the distribution functions of U_1, U_2 , and U_3 , for $t > 0$, as $F_{U_1}(t, \alpha_1) = (1 - e^{-t})^{\alpha_1}$, $F_{U_2}(t, \alpha_2) = (1 - e^{-t})^{\alpha_2}$, and $F_{U_3}(t; \lambda) = (1 - e^{-\lambda t})$, respectively. Consider a bivariate random variable (X, Y) , such that $X = \min\{U_1, U_3\}$ and $Y = \min\{U_2, U_3\}$. It may be mentioned that although U_1, U_2 , and U_3 belong to proportional reversed hazard class, the authors considered the minimization approach. Due to that reason the joint PDF of X and Y is not a very a compact form. The joint SF and the joint PDF of X and Y also can be derived along the same way, as we have mentioned in the previous section. The joint SF of X and Y in this case becomes

$$S_{X,Y}(x, y) = P(X > x, Y > y) \\ = \begin{cases} e^{-\lambda x}(1 - (1 - e^{-x})^{\alpha_1})(1 - (1 - e^{-y})^{\alpha_2}) & \text{if } 0 < y < x < \infty \\ e^{-\lambda y}(1 - (1 - e^{-x})^{\alpha_1})(1 - (1 - e^{-y})^{\alpha_2}) & \text{if } 0 < x \leq y < \infty, \end{cases}$$

and the joint PDF can be obtained as

Here

$$f_{X,Y}(x, y) = \begin{cases} f_1(x, y) & \text{if } 0 < y < x < \infty, \\ f_2(x, y) & \text{if } 0 < x < y < \infty, \\ f_0(x) & \text{if } 0 < x = y < \infty. \end{cases}$$

$$f_1(x, y) = \alpha_2 e^{-(\lambda x + y)} (1 - e^{-y})^{\alpha_2 - 1} (\lambda - \lambda(1 - e^{-x})^{\alpha_1} + \alpha_1 e^{-x}(1 - e^{-x})^{\alpha_1 - 1}) \\ f_2(x, y) = \alpha_1 e^{-(\lambda y + x)} (1 - e^{-x})^{\alpha_1 - 1} (\lambda - \lambda(1 - e^{-y})^{\alpha_2} + \alpha_2 e^{-y}(1 - e^{-y})^{\alpha_2 - 1}) \\ f_0(x) = \lambda e^{-\lambda x} (1 - (1 - e^{-x})^{\alpha_1}) (1 - (1 - e^{-y})^{\alpha_2}).$$

Therefore, the SF and X and Y become

$$S_X(x) = P(X > x) = e^{-\lambda x} (1 - (1 - e^{-x})^{\alpha_1}) \quad \text{and} \quad S_Y(y) = P(Y > y) = e^{-\lambda y} (1 - (1 - e^{-y})^{\alpha_2}),$$

respectively. It should also be mentioned that the copula associated with the Sarhan-Balakrishnan bivariate distribution cannot be obtained in an explicit form.

where $z = \max\{x, y\}$. Hence the marginal survival function can be obtained as

$$P(X > x) = (1 - (1 - e^{-\lambda x})^{\alpha_1})(1 - (1 - e^{-\lambda z})^{\alpha_3}) \\ P(Y > y) = (1 - (1 - e^{-\lambda y})^{\alpha_2})(1 - (1 - e^{-\lambda z})^{\alpha_3}).$$

36.7.2 Modified Sarhan-Balakrishnan Bivariate Distribution

Kundu and Gupta [33] proposed a modified version of the Sarhan-Balakrishnan bivariate distribution, and the latter can be obtained as a special case of the former. The idea is as follows: Suppose U_1 , U_2 , and U_3 are three independent random variables, such that for $t > 0$, the CDF of U_1 , U_2 , and U_3 are $F_{U_1}(t; \alpha_1, \lambda) = (1 - e^{-\lambda t})^{\alpha_1}$, $F_{U_2}(t; \alpha_2, \lambda) = (1 - e^{-\lambda t})^{\alpha_2}$, and $F_{U_3}(t; \alpha_3, \lambda) = (1 - e^{-\lambda t})^{\alpha_3}$, respectively. In this case also consider a bivariate random variable (X, Y) , such that $X = \min\{U_1, U_3\}$ and $Y = \min\{U_2, U_3\}$. Clearly, the modified Sarhan-Balakrishnan bivariate distribution is more flexible than the Sarhan-Balakrishnan bivariate distribution.

The joint survival function in this case can be obtained as

$$S_{X,Y}(x, y) = P(X > x, Y > y) \\ = (1 - (1 - e^{-\lambda x})^{\alpha_1})(1 - (1 - e^{-\lambda y})^{\alpha_2}) \\ (1 - (1 - e^{-\lambda z})^{\alpha_3}),$$

Along the same line as before, the joint PDF of the modified Sarhan-Balakrishnan bivariate distribution can be obtained as

$$f_{X,Y}(x, y) = \begin{cases} f_1(x, y) & \text{if } 0 < y < x < \infty, \\ f_2(x, y) & \text{if } 0 < x < y < \infty, \\ f_0(x) & \text{if } 0 < x = y < \infty, \end{cases}$$

where

$$f_1(x, y) = f(x; \alpha_1, \lambda)[f(y; \alpha_2, \lambda) + f(y; \alpha_3, \lambda) \\ - f(y; \alpha_2 + \alpha_3, \lambda)] \\ f_2(x, y) = f(y; \alpha_2, \lambda)[f(x; \alpha_1, \lambda) + f(x; \alpha_3, \lambda) \\ - f(x; \alpha_1 + \alpha_3, \lambda)]$$

$$f_0(x) = \frac{\alpha_1}{\alpha_1 + \alpha_3} f(x; \alpha_1 + \alpha_3, \lambda) + \frac{\alpha_2}{\alpha_2 + \alpha_3} f(x; \alpha_2 + \alpha_3, \lambda) \\ - \frac{\alpha_1 + \alpha_2}{\alpha_1 + \alpha_2 + \alpha_3} f(x; \alpha_1 + \alpha_2 + \alpha_3, \lambda).$$

Here

$$f(x; \alpha, \lambda) = \begin{cases} \alpha \lambda e^{-\lambda x} (1 - e^{-\lambda x})^{\alpha-1} & \text{if } x > 0 \\ 0 & \text{if } x \leq 0. \end{cases}$$

In this also the copula associated with the modified Sarhan-Balakrishnan bivariate distribution cannot be obtained in a compact form. Note that this idea can be extended for the proportional reversed hazard classes also. The necessary EM algorithm also can be developed for this model along the same line.

36.7.3 Bivariate Weibull-Geometric Distribution

As it has been mentioned before, Marshall and Olkin [45] proposed a method to introduce a parameter in a family of

distributions. It introduces an extra parameter to a model; hence it brings more flexibility. The model has some interesting physical interpretations also. The basic idea is very simple, and it can be defined as follows. Suppose $\{X_n; n = 1, 2, \dots\}$ is a sequence of i.i.d. random variables and N is a geometric random variable with parameter $0 < p \leq 1$ having probability mass function

$$P(N = n) = p(1 - p)^{n-1}; \quad n = 1, 2, \dots \quad (36.31)$$

Here, X'_n 's and N are independently distributed. Consider the random variables,

$$Y = \min\{X_1, \dots, X_N\}.$$

We denote $S_X(\cdot)$, $S_Y(\cdot)$, $f_X(\cdot)$, and $f_Y(\cdot)$ as the survival function of X , survival function of Y , PDF of X_1 , and PDF of Y , respectively. The CDF of Y becomes

$$\begin{aligned} S_Y(y) &= P(Y > y) = P(X_1 > y, \dots, X_N > y) = \sum_{n=1}^{\infty} P(X_1 > y, \dots, X_n > y | N = n) P(N = n) \\ &= p \sum_{n=1}^{\infty} S_X^n(y) (1 - p)^{n-1} = \frac{p S_X(y)}{1 - (1 - p) S_X(y)}. \end{aligned}$$

Therefore, the PDF of Y becomes

$$f_Y(y) = \frac{p f_X(y)}{(1 - (1 - p) S_X(y))^2}.$$

If X_1 is an exponential random variable with mean $1/\lambda$, then Y is called Marshall-Olkin exponential (MOE) random variable. The PDF of a MOE distribution with parameter λ and p , MOE(p, λ), is

$$f_Y(y; p, \lambda) = \frac{p \lambda e^{-\lambda y}}{(1 - (1 - p) e^{-\lambda y})^2}; \quad y > 0. \quad (36.32)$$

When $p = 1$, it becomes an exponential random variable. From the PDF (36.32), it is clear that MOE is a weighted exponential random variable, where the weight function is

$$w(y) = \frac{p}{(1 - (1 - p) e^{-\lambda y})^2}; \quad y > 0.$$

Hence, the two-parameter MOE distribution is more flexible than the one parameter exponential distribution. Similarly, if X_1 is a Weibull random variable with the shape parameter $\alpha > 0$ and scale parameter $\lambda > 0$, with the PDF

$$f_X(x; \alpha, \lambda) = \alpha \lambda x^{\alpha-1} e^{-\lambda x^\alpha}; \quad x > 0$$

then the associated Y is said to have Marshall-Olkin Weibull (MOWE) distribution. The MOWE has the PDF

$$f_Y(y; p, \alpha, \lambda) = \frac{p \alpha \lambda y^{\alpha-1} e^{-\lambda y^\alpha}}{(1 - (1 - p) e^{-\lambda y^\alpha})^2}.$$

It is immediate that the MOWE distribution is a weighted Weibull distribution, and in this case the weight function is

$$w(y) = \frac{p}{(1 - (1 - p) e^{-\lambda y^\alpha})^2}; \quad y > 0.$$

The three-parameter MOWE distribution is more flexible than the two-parameter Weibull distribution. Marshall and Olkin [45] developed several properties of the MOE and MOWE distributions. Although Marshall and Olkin [45] did not develop any estimation procedure, very effective EM algorithm can be developed as it has been obtained in case of the corresponding bivariate model by Kundu and Gupta [36].

In the same paper Marshall and Olkin [45] mentioned how their method can be extended to the multivariate case also, although they did not provide much detail. Moreover, the extension to the bivariate or multivariate case may not be unique. Kundu and Gupta [36] first used the method of Marshall and Olkin [45] to extend MOBW distribution and called it as the bivariate Weibull-Geometric (BWG) distribution.

It has five parameters, and MOBW can be obtained as a special case of BWG distribution. The details are given below.

Suppose $\{(X_{1n}, X_{2n}); n = 1, 2, \dots\}$ is a sequence of i.i.d. random variables with parameters $\alpha_1, \alpha_2, \alpha_3, \theta$ as given in (36.17) and

$$Y_1 = \min\{X_{11}, \dots, X_{1N}\} \quad \text{and} \quad Y_2 = \min\{X_{21}, \dots, X_{2N}\},$$

then (Y_1, Y_2) is said to have BWG distribution.

The joint survival function of (Y_1, Y_2) becomes

$$P(Y_1 > y_1, Y_2 > y_2) = \sum_{n=1}^{\infty} P^n(X_{11} > y_1, X_{21} > y_2) \quad \text{where}$$

$$p(1-p)^{n-1}$$

$$g_1(y_1, y_2) = \frac{p\theta^2 y_1^{\theta-1} y_2^{\theta-1} (\alpha_3 + \alpha_1) \alpha_2 e^{-(\alpha_1 + \alpha_3) y_1^\theta - \alpha_2 y_2^\theta} (1 + (1-p)e^{-(\alpha_1 + \alpha_3) y_1^\theta - \alpha_2 y_2^\theta})}{(1 - (1-p)e^{-(\alpha_3 + \alpha_1) y_1^\theta - \alpha_2 y_2^\theta})^3}$$

$$g_2(y_1, y_2) = \frac{p\theta^2 y_1^{\theta-1} y_2^{\theta-1} (\alpha_3 + \alpha_2) \alpha_1 e^{-(\alpha_2 + \alpha_3) y_2^\theta - \alpha_1 y_1^\theta} (1 + (1-p)e^{-(\alpha_2 + \alpha_3) y_2^\theta - \alpha_1 y_1^\theta})}{(1 - (1-p)e^{-(\alpha_3 + \alpha_2) y_2^\theta - \alpha_1 y_1^\theta})^3}$$

$$g_0(y) = \frac{p\theta y^{\theta-1} \alpha_3 e^{-(\alpha_1 + \alpha_2 + \alpha_3) y^\theta}}{(\alpha_1 + \alpha_2 + \alpha_3)(1 - (1-p)e^{-(\alpha_1 + \alpha_2 + \alpha_3) y^\theta})^2}$$

$$= \frac{pP(X_{11} > y_1, X_{21} > y_2)}{1 - (1-p)P(X_{11} > y_1, X_{21} > y_2)}$$

$$= \begin{cases} \frac{pe^{-\alpha_1 y_1^\theta} e^{-(\alpha_2 + \alpha_3) y_2^\theta}}{1 - (1-p)e^{-\alpha_1 y_1^\theta} e^{-(\alpha_2 + \alpha_3) y_2^\theta}} & \text{if } y_1 \leq y_2 \\ \frac{pe^{-(\alpha_1 + \alpha_3) y_1^\theta} e^{-\alpha_2 y_2^\theta}}{1 - (1-p)e^{-(\alpha_1 + \alpha_3) y_1^\theta} e^{-\alpha_2 y_2^\theta}} & \text{if } y_1 > y_2. \end{cases}$$

The joint PDF of (Y_1, Y_2) can be obtained as

$$g(y_1, y_2) = \begin{cases} g_1(y_1, y_2) & \text{if } y_1 > y_2 \\ g_2(y_1, y_2) & \text{if } y_1 < y_2 \\ g_0(y) & \text{if } y_1 = y_2 = y, \end{cases} \quad (36.33)$$

The surface plot of the absolute continuous part of BWG for different parameter values is provided in Fig. 36.6. It is clear from Fig. 36.6 that BWG is more flexible than the MOBW distribution.

From the joint PDF of (Y_1, Y_2) it can be seen that

$$P(Y_1 = Y_2) = \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} > 0,$$

hence it has a singular component. The marginals of Y_1 and Y_2 are geometric-Weibull distribution as defined by Marshall and Olkin [45]. It may be mentioned that when $\theta = 1$, BWG distribution becomes the MOBW distribution. Therefore, MOBW distribution can be obtained as a special case of the BWG distribution. Estimation of the unknown five parameters for the BWG distribution based on a bivariate set of data set is a challenging problem. The MLEs of the unknown parameters can be obtained by solving a five-dimensional optimization problem. To avoid that, Kundu and Gupta [36] proposed a very efficient EM algorithm, which involves solving a one-dimensional optimization problem at each ‘‘E’’-step of the EM algorithm. In the same paper Kundu and Gupta [36] developed the testing problem so that based of the observed data it should be possible to test whether it is coming from BWG or MOBW model. This model can be used for modeling dependent competing risks data.

36.7.4 Bivariate PHM-Geometric Distribution

Although Kundu and Gupta [36] proposed the Marshall-Olkin method for the MOBW distribution, it can be extended for a general bivariate PHM as provided in (36.11). Similar to the bivariate Weibull-geometric model, more general the bivariate PHM-geometric distribution can be obtained as follows: Suppose $\{(X_{1n}, X_{2n}); n = 1, 2, \dots\}$ is a sequence of i.i.d. random variables with parameters $\alpha_1, \alpha_2, \alpha_3$ as follows:

$$S_{X,Y}(x, y) = \begin{cases} [S_B(x)]^{\alpha_1} [S_B(y)]^{\alpha_2 + \alpha_3} & \text{if } x < y \\ [S_B(x)]^{\alpha_1 + \alpha_3} [S_B(y)]^{\alpha_2} & \text{if } x \geq y. \end{cases} \quad (36.34)$$

Here $S_B(\cdot)$ is the base line survival function. It may depend on some parameter, but we do not make it explicit. Further, N is a geometric random variable with parameter $0 < p \leq 1$ having probability mass function as given in (36.31), and it is independent of $\{(X_{1n}, X_{2n}); n = 1, 2, \dots\}$. Consider the random variables,

$$Y_1 = \min\{X_{11}, \dots, X_{1N}\} \quad \text{and} \quad Y_2 = \min\{X_{21}, \dots, X_{2N}\},$$

then (Y_1, Y_2) is said to have bivariate PHM-Geometric (PHMG) distribution. The joint survival function of the bivariate PHMG becomes

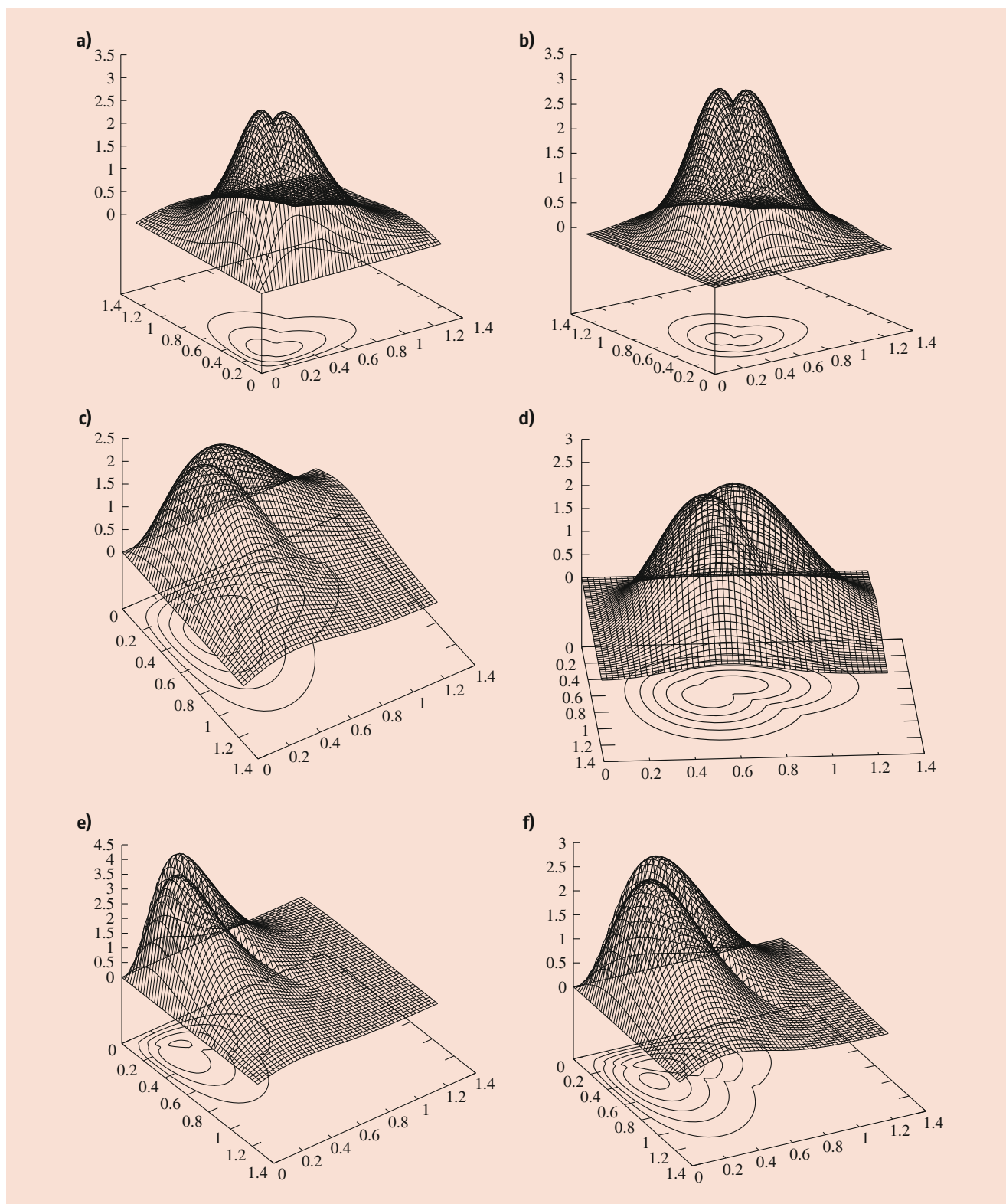


Fig. 36.6 PDF plots of $MOBE(\alpha_1, \alpha_2, \alpha_3, \theta, p)$ distribution for different $(\alpha_1, \alpha_2, \alpha_3, \theta, p)$ values: (a) (2.0,2.0,2.0,2.0,0.5) (b) (2.0,2.0,2.0,3.0,0.5) (c) (2.0,2.0,2.0,2.0,0.8) (d) (2.0,2.0,2.0,3.0,0.8) (e) (2.0,3.0,2.0,2.0,0.5) (f) (2.0,3.0,2.0,2.0,0.8)

$$\begin{aligned}
 P(Y_1 > y_1, Y_2 > y_2) &= \sum_{n=1}^{\infty} P^n(X_{11} > y_1, X_{21} > y_2) \\
 &= \frac{p(1-p)^{n-1}}{1 - (1-p)P(X_{11} > y_1, X_{21} > y_2)} \\
 &= \begin{cases} \frac{p[S_B(y_1)]^{\alpha_1}[S_B(y_2)]^{\alpha_2+\alpha_3}}{1-(1-p)[S_B(y_1)]^{\alpha_1}[S_B(y_2)]^{\alpha_2+\alpha_3}} & \text{if } y_1 \leq y_2 \\ \frac{p[S_B(y_1)]^{\alpha_1+\alpha_3}[S_B(y_2)]^{\alpha_2}}{1-(1-p)[S_B(y_1)]^{\alpha_1+\alpha_3}[S_B(y_2)]^{\alpha_2}} & \text{if } y_1 > y_2. \end{cases}
 \end{aligned}$$

If we denote $f_B(\cdot)$ as the base line PDF, then the joint PDF of the bivariate PHMG can be written as

$$g(y_1, y_2) = \begin{cases} g_1(y_1, y_2) & \text{if } y_1 > y_2 \\ g_2(y_1, y_2) & \text{if } y_1 < y_2 \\ g_0(y) & \text{if } y_1 = y_2 = y, \end{cases}$$

where

$$\begin{aligned}
 g_1(y_1, y_2) &= \frac{c_1 f_B(y_1)[S_B(y_1)]^{\alpha_1+\alpha_3-1} f_B(y_2)[S_B(y_2)]^{\alpha_2-1} (1 + (1-p)[S_B(y_1)]^{\alpha_1+\alpha_3}[S_B(y_2)]^{\alpha_2})}{(1 + (1-p)[S_B(y_1)]^{\alpha_1+\alpha_3}[S_B(y_2)]^{\alpha_2})^3} \\
 g_2(y_1, y_2) &= \frac{c_2 f_B(y_1)[S_B(y_1)]^{\alpha_1-1} f_B(y_2)[S_B(y_2)]^{\alpha_2+\alpha_3-1} (1 + (1-p)[S_B(y_1)]^{\alpha_1}[S_B(y_2)]^{\alpha_2+\alpha_3})}{(1 + (1-p)[S_B(y_1)]^{\alpha_1}[S_B(y_2)]^{\alpha_2+\alpha_3})^3} \\
 g_0(y) &= \frac{p f_B(y) \alpha_3 [S_B(y)]^{\alpha_1+\alpha_2+\alpha_3-1}}{(\alpha_1 + \alpha_2 + \alpha_3)(1 - (1-p)[S_B(y)]^{\alpha_1+\alpha_2+\alpha_3})^2}
 \end{aligned}$$

$$c_1 = p(\alpha_3 + \alpha_1)\alpha_2, \quad c_2 = p(\alpha_3 + \alpha_2)\alpha_1.$$

Clearly, bivariate PHMG is more flexible than the BWG distribution. When the base line distribution is Weibull with the scale parameter one, and the shape parameter θ , then the bivariate PHMG becomes the BWG distribution. This model can also be used for analyzing dependent competing risks data. It will be interesting to develop both classical and Bayesian inference procedures for this distribution. More work is needed in this direction.

36.7.5 Bivariate GE-Geometric Distribution

In this section first we introduce Marshall-Olkin GE (MOGE) distribution similar to the MOE and MOWE distribution. The

idea is very similar, and it can be defined as follows. Suppose $\{X_n; n = 1, 2, \dots\}$ is a sequence of i.i.d. $GE(\alpha, \lambda)$ random variables and N is a geometric random variable with the PMF as given in (36.31). Moreover, N and $\{X_n; n = 1, 2, \dots\}$ are independently distributed. Let us define a new random variable Y as

$$Y = \max\{X_1, \dots, X_N\}.$$

The random variable Y is said to have GE-Geometric (GEG) distribution with parameter p, α, λ , and it will be denoted by $GEG(p, \alpha, \lambda)$.

The CDF of Y for $y > 0$ can be obtained as

$$\begin{aligned}
 F_Y(y; \alpha, \lambda) &= P(Y \leq y) = P(X_1 \leq y, \dots, X_N \leq y) = p \sum_{n=1}^{\infty} P^n(X_1 \leq y)(1-p)^{n-1} \\
 &= \frac{p(1 - e^{-\lambda x})^\alpha}{1 - (1-p)(1 - e^{-\lambda x})^\alpha}.
 \end{aligned}$$

The corresponding PDF becomes

$$f_Y(y; p, \alpha, \lambda) = \frac{p\alpha\lambda e^{-\lambda x}(1 - e^{-\lambda x})^{\alpha-1}}{(1 - (1-p)(1 - e^{-\lambda x})^\alpha)^2} \quad \text{for } y > 0,$$

and zero, otherwise.

Now we can define bivariate GEG (BGEG) along the same line as before. Suppose $\{(X_{1n}, X_{2n}); n = 1, 2, \dots\}$ is a

sequence of i.i.d. random variables with parameters $\beta_1, \beta_2, \beta_3, \lambda$ having the joint PDF as given in (36.20) and

$$Y_1 = \max\{X_{11}, \dots, X_{1N}\} \quad \text{and} \quad Y_2 = \max\{X_{21}, \dots, X_{2N}\},$$

then (Y_1, Y_2) is said to have BGEG distribution with parameters $\beta_1, \beta_2, \beta_3, \lambda, p$.

The joint CDF of (Y_1, Y_2) becomes

$$\begin{aligned}
P(Y_1 \leq y_1, Y_2 \leq y_2) &= \sum_{n=1}^{\infty} P^n(X_{11} \leq y_1, X_{21} \leq y_2) \\
&\quad p(1-p)^{n-1} \\
&= \frac{pP(X_{11} \leq y_1, X_{21} \leq y_2)}{1 - (1-p)P(X_{11} \leq y_1, X_{21} \leq y_2)} \\
&= \begin{cases} \frac{p(1-e^{-\lambda y_1})^{\beta_1} (1-e^{-\lambda y_2})^{\beta_2+\beta_3}}{1-(1-p)(1-e^{-\lambda y_1})^{\beta_1} (1-e^{-\lambda y_2})^{\beta_2+\beta_3}} & \text{if } y_1 > y_2 \\ \frac{p(1-e^{-\lambda y_1})^{\beta_1+\beta_3} (1-e^{-\lambda y_2})^{\beta_2}}{1-(1-p)(1-e^{-\lambda y_1})^{\beta_1+\beta_3} (1-e^{-\lambda y_2})^{\beta_2}} & \text{if } y_1 \leq y_2. \end{cases}
\end{aligned}$$

It will be interesting to develop different properties of this distribution similar to the BWG distribution as it has been obtained by Kundu and Gupta [36]. Moreover, classical and Bayesian inference need to be developed for analyzing bivariate data sets with ties. This model also can be used for analyzing dependent complementary risks data. More work is needed along this direction.

36.8 Conclusions

In this chapter we have considered the class of bivariate distributions with a singular component. Marshall and Olkin [44] first introduced the bivariate distribution with singular component based on exponential distributions. Since then an extensive amount of work has taken place in this direction. In this chapter we have provided a comprehensive review of all the different models. It is observed that there are mainly two main approaches available in the literature to define bivariate distributions with singular components, and they produce two broad classes of bivariate distributions with singular components. We have shown that these two classes of distributions are related through their copulas under certain restriction. We have provided very general EM algorithms which can be used to compute the MLEs of the unknown parameters in these two cases. We have provided the analysis of one data set to show how the EM algorithms perform in real life.

There are several open problems associated with these models. Although in this chapter we have mainly discussed different methods for bivariate distributions, all the methods can be generalized for the multivariate distribution also. Franco and Vivo [20] first obtained the multivariate Sarhan-Balakrishnan distribution and developed several properties, although no inference procedures were developed. Sarhan et al. [60] provided the multivariate generalized failure rate distribution, and Franco, Kundu, and Vivo [19] developed multivariate modified Sarhan-Balakrishnan distribution. In both these papers, the authors developed different properties and very efficient EM algorithms for computing the maximum likelihood estimators of the unknown parameters. More general multivariate distributions with proportional reversed hazard marginals can be found in Kundu, Franco, and Vivo [38]. Note that all these models can be very useful to analyze

dependent competing risks model with multiple causes. No work has been done along that like; more work is needed along these directions.

References

1. Al-Hussaini, E.K., Ahsanullah, M.: Exponentiated Distributions. Atlantis Press, Paris, France (2015)
2. Arnold, B.C.: Parameter estimation for a multivariate exponential distribution. *J. Am. Stat. Assoc.* **63**, 848–852 (1968)
3. Azzalini, A.A., Capitanio, A.: The Skew-Normal and Related Families. Cambridge University Press, Cambridge, UK (2014)
4. Azzalini, A.A., Dalla Valle, A.: The multivariate skew normal distribution. *Biometrika* **83**:715–726 (1996)
5. Balakrishnan, N., Basu, A.: Exponential Distribution: Theory, Methods and Applications. CRC Press, New York (1996)
6. Balakrishnan, N., Kundu, D.: Birnbaum-Saunders distribution: A review of models, analysis and applications (with discussions). *Appl. Stochast. Models Bus. Ind.* **35**(1), 4–132 (2019)
7. Balakrishnan, N., Lai, C.: Continuous Bivariate Distributions, 2nd edn. Springer, New York (2009)
8. Baxter, L.A., Rachev, S.T.: The stability of a characterization of the bivariate Marshall-Olkin distribution. *J. Math. Anal. Appl.* **160**, 563–573 (1991)
9. Barreto-Souza, W., Lemonte, A.J.: Bivariate Kumaraswamy distribution: properties and a new method to generate bivariate classes. *Statistics* **47**, 1321–1342 (2013)
10. Begum, A.A., Khan, A.H.: Concomitants of order statistics from Marshall and Olkin's bivariate Weibull distribution. *Calcutta Stat. Assoc. Bull.* **50**, 197–198 (2000)
11. Bemis, B., Bain, L.J., Higgins, J.J.: Estimation and hypothesis testing for the parameters of a bivariate exponential distribution. *J. Am. Stat. Assoc.* **67**, 927–929 (1972)
12. Bhattacharyya, G.K., Johnson, R.A.: Maximum likelihood estimation and hypothesis testing in the bivariate exponential model of Marshall and Olkin (1971). <https://handle.dtic.mil/100.2/AD737530>
13. Block, H.W., Savits, T.H., Singh, H.: On the reversed hazard rate function. *Probab. Eng. Inf. Sci.* **12**, 69–90 (1998)
14. Boland, P.J.: An arrangement increasing property of the Marshall-Olkin bivariate exponential. *Stat. Probab. Lett.* **37**, 167–170 (1998)
15. Cai, J., Shi, Y., Liu, B.: Analysis of incomplete data in the presence of dependent competing risks from Marshall-Olkin bivariate Weibull distribution under progressive hybrid censoring. *Commun. Stat. Theory Methods* **46**, 6497–6511 (2017)
16. Cox, D.R.: Regression models and life-tables. *J. Roy. Stat. Soc. B* **34**, 187–220 (1972)
17. Cox, D.R., Oakes, D.: Analysis of Survival Data. Chapman & Hall, New York (1984)
18. Feizjavadian, S.H., Hashemi, R.: Analysis of dependent competing risks in the presence of progressive hybrid censoring using Marshall-Olkin bivariate Weibull distribution. *Comput. Stat. Data Anal.* **82**, 19–34 (2015)
19. Franco, M., Kundu, D., Juana-Maria, V.: Multivariate extension of modified Sarhan-Balakrishnan bivariate distribution. *J. Stat. Plann. Infer.* **141**, 3400–3412 (2011)
20. Franco, M., Juana-Maria, V.: A multivariate extension of Sarhan and Balakrishnan's bivariate distribution and its aging and dependence properties. *J. Multivariate Anal.* **101**, 494–499 (2009)
21. Gupta, R.D., Kundu, D.: Generalized exponential distribution. *Aust. New Zealand J. Stat.* **41**, 173–188 (1999)
22. Gupta, R.D., Kundu, D.: A new class of weighted exponential distributions. *Statistics* **43**, 621–634 (2009)

23. Gupta, R.C., Kirmani, S.N.U.A., Balakrishnan, N.: On a class of generalized Marshall-Olkin bivariate distributions and some reliability characteristics. *Probab. Eng. Inf. Sci.* **27**, 261–275 (2013)
24. Jamalizadeh, A., Kundu, D.: Weighted Marshall-Olkin bivariate exponential distribution. *Statistics* **47**, 917–928 (2013)
25. Jose, K.K., Ristic, M.M., Joseph, A.: Marshall-Olkin bivariate Weibull distributions and processes. *Statistical Papers* **52**, 789–798 (2011)
26. Karlis, D.: ML estimation for multivariate shock models via an EM algorithm. *Anna. Institut. Stat. Math.* **55**, 817–830 (2003)
27. Kotz, S., Balakrishnan, N., Johnson, N.: *Continuous Multivariate Distributions*, 2nd edn. Wiley, New York (2000)
28. Kundu, D.: Bivariate geometric (maximum) generalized exponential distribution. *J. Data Sci.* **13**, 693–712 (2015)
29. Kundu, D.: Multivariate geometric skew normal distribution. *Statistics* **51**, 1377–1397 (2017)
30. Kundu, D., Dey, A.K.: Estimating the parameters of the Marshall Olkin Bivariate Weibull distribution by EM Algorithm. *Comput. Stat. Data Anal.* **53**, 956–965 (2009)
31. Kundu, D., Gupta, R.D.: Characterizations of the proportional (reversed) hazard class. *Commun. Stat. Theory Methods* **38**(12), 3095–3102 (2004)
32. Kundu, D., Gupta, R.D.: Bivariate generalized exponential distribution. *J. Multivariate Anal.* **100**, 581–593 (2009)
33. Kundu, D., Gupta, R.D.: Modified Sarhan-Balakrishnan singular bivariate distribution. *J. Stat. Plann. Infer.* **140**, 526–538 (2010a)
34. Kundu, D., Gupta, R.D.: A class of bivariate models with proportional reversed hazard marginals. *Sankhya B* **72**, 236–253 (2010b)
35. Kundu, D., Gupta, A.K.: Bayes estimation for the Marshall-Olkin bivariate Weibull distribution. *Comput. Stat. Data Anal.* **57**, 271–281 (2013)
36. Kundu, D., Gupta, A.K.: On bivariate Weibull Geometric distribution. *J. Multivariate Anal.* **123**(1), 19–29 (2014)
37. Kundu, D., Gupta, A.K.: On bivariate inverse Weibull distribution. *Brazilian J. Probab. Stat.* **31**, 275–302 (2017)
38. Kundu, D., Franco, M., Juana-Maria, V.: Multivariate distributions with proportional reversed Hazard marginals. *Comput. Stat. Data Anal.* **77**, 98–212 (2014)
39. Lai, C.-D., Lin, G.D., Govindaraju, K., Pirikahu, S.A.: A simulation study on the correlation structure of Marshall-Olkin bivariate Weibull distribution. *J. Stat. Comput. Simul.* **87**, 156–170 (2017)
40. Louis, T.A.: Finding the observed information matrix when using the EM algorithm. *J. Roy. Stat. Soc. B* **44**, 226–233 (1982)
41. Lu, J.-C.: Weibull extension of the Freund and Marshall-Olkin bivariate exponential model. *IEEE Trans. Reliab.* **38**, 615–619 (1989)
42. Lu, J.-C.: Bayes parameter estimation for the bivariate Weibull model of Marshall-Olkin for censored data. *IEEE Trans. Reliab.* **41**, 608–615 (1992)
43. Lehmann, E.L.: The power of rank test. *Anna. Math. Stat.* **24**, 23–42 (1953)
44. Marshall, A.W., Olkin, I.: A multivariate exponential distribution. *J. Am. Stat. Assoc.* **62**, 30–44 (1967)
45. Marshall, A.W., Olkin, I.: A new method of adding a parameter to a family of distributions with application to the exponential and Weibull families. *Biometrika* **84**, 641–652 (1997)
46. Meintanis, S.G.: Test of fit for Marshall-Olkin distributions with applications. *J. Stat. Plann. Infer.* **137**, 3954–3963 (2007)
47. Mondal, S., Kundu, D.: A bivariate inverse Weibull distribution and its application in complementary risks model. *J. Appl. Stat.* (2019, to appear)
48. Mudholkar, G.S., Srivastava, D.K., Freimer, M.: The exponentiated Weibull family: a reanalysis of the bus-motor-failure data. *Technometrics* **37**, 436–445 (1995)
49. Muhammed, H.: Bivariate inverse Weibull distribution. *J. Stat. Comput. Simul.* **86**, 2335–2345 (2016)
50. Muliere, P., Scarsini, M.: Characterization of a Marshall-Olkin type class of distributions. *Anna. Inst. Stat. Math.* **39**, 429–441 (1987)
51. Nadarajah, S.: Marshall and Olkin's distributions. *Acta Appl. Math.* **103**, 87–100 (2008)
52. Nadarajah, S.: The exponentiated exponential distribution; a survey. *Adv. Stat. Anal.* **95**, 219–251 (2011)
53. Nelsen, R.B.: *An Introduction to Copula*, 2nd edn. Springer, New York (2006)
54. Pena, E.A., Gupta, A.K.: Bayes estimation for the Marshall-Olkin exponential distribution. *J. Roy. Stat. Soc. B* **52**, 379–389 (1990)
55. Proschan, F., Sullo, P.: Estimating the parameters of a multivariate exponential distribution. *J. Am. Stat. Assoc.* **71**, 465–472 (1976)
56. Rinne, H.: *The Weibull Distribution: A Handbook*, 1st edn. Chapman and Hall/CRC, New York (2008)
57. Ryu, K.: An extension of Marshall and Olkins bivariate exponential distribution. *J. Am. Stat. Assoc.* **88**, 1458–1465 (1993)
58. Sarhan, A.M., Balakrishnan, N.: A new class of bivariate distribution and its mixture. *J. Multivariate Anal.* **98**, 1508–1527 (2007)
59. Sarhan, A.M., Kundu, D.: Generalized linear failure rate distribution. *Commun. Stat. Theory Methods* **38**, 642–660 (2009)
60. Sarhan, A.M., Hamilton, D.C., Smith, B., Kundu, D.: The bivariate generalized linear failure rate distribution and its multivariate extension. *Comput. Stat. Data Anal.* **55**, 644–654 (2011)
61. Shen, Y., Xu, A.: On the dependent competing risks using Marshall-Olkin bivariate Weibull model: Parameter estimation with different methods. *Commun. Stat. Theory Methods* **47**, 5558–5572 (2018)
62. Surles, J.G., Padgett, W.J.: Inference for reliability and stress-strength for a scaled Burr Type X distribution. *Lifetime Data Anal.* **7**, 187–200 (2001)
63. Therneau, T.M., Grambsch, P.M.: *Modeling Survival Data: Extending the Cox Model*. Springer, New York (2000)



Debasis Kundu Professor Debasis Kundu received his B.Stat and M.Stat from the Indian Statistical Institute in 1982 and 1984, respectively, M.A. in Mathematics from the University of Pittsburgh in 1985 and Ph.D. from the Pennsylvania State University in 1989 under the guidance of Professor C.R. Rao. After finishing his Ph.D. he joined The University of Texas at Dallas as an Assistant Professor before joining Indian Institute of Technology Kanpur in 1990. He served as the Head of the Department of Mathematics and Statistics, IIT Kanpur from 2011 to 2014 and the Dean of Faculty Affairs at IIT Kanpur from 2019 to 2022. Professor Kundu is holding an Endowed Chair Professor post of the Institute since 2008.

Professor Kundu works on different areas of Statistics. His major research interests are on Statistical Signal Processing, Lifetime Data Analysis and Statistical Computing. He has published more than 325 research papers in different well reputed Statistical and Signal Processing journals like *IEEE Transaction on Signal Processing*; *Signal Processing*; *Circuits, Systems and Signal Processing*; *Technometrics*; *Annals of the Institute of Statistical Mathematics*; *Statistica Sinica*; *Scandinavian Journal of Statistics*; *Journal of Multivariate Analysis*; *Computational Statistics and Data Analysis*;

Journal of Statistical Planning and Inference; IEEE Transactions on Reliability etc. His work has been well cited by different researchers. He has more than 16,238 google scholar citations with H index 63. He has co-authored two research monographs on Statistical Signal Processing and on Step-Stress Models, and co-edited one book on Statistical Computing. He is a Fellow of the National Academy of Sciences, India, and a Fellow of the Indian Society

of Probability and Statistics. He is currently the Editor-in-Chief of the Journal of the Indian Society of Probability and Statistics, and in the editorial boards of Sankhya, Ser. B, Indian Journal of Pure and Applied Mathematics and Journal of Statistical Theory and Practice. He was in the editorial boards of the IEEE Transactions on Reliability, Journal of Distribution Theory, Communications in Statistics - Theory and Methods, Communications in Statistics - Simulation and Computation.



Ashis Kumar Chakraborty , Soumen Dey , Poulami Chakraborty, and Aleena Chanda

Contents

37.1	Introduction	764	37.11	Model Comparison	782
37.2	Conditional Probability	764	37.11.1	Akaike Information Criterion (AIC)	782
37.3	Some Examples	764	37.11.2	Deviance Information Criterion (DIC)	782
37.3.1	Prosecutor's Fallacy	764	37.11.3	Widely Applicable Information Criterion (WAIC)	782
37.3.2	Monty Hall Problem	765	37.11.4	Out-of-Sample Validation	783
37.3.3	Types of Prior	765	37.11.5	Bayesian Cross-Validation	783
37.3.4	Reference Prior	766	37.11.6	Posterior Predictive Loss	784
37.4	Hierarchical Bayes	767	37.11.7	Bayes Factor	784
37.5	Empirical Bayes	767	37.11.8	Bayesian Model Averaging	786
37.5.1	Non-parametric Empirical Bayes	768	37.11.9	Bayesian Information Criterion (BIC)	786
37.5.2	Parametric Empirical Bayes	768	37.12	Some Applications of Bayesian Methods in Different Fields of Research	787
37.6	Point Estimation	768	37.12.1	Example from Statistical Ecology	787
37.6.1	Univariate Case	768	37.12.2	Process Control	788
37.6.2	Multivariate Case	769	37.12.3	Reliability	789
37.7	Bayesian Interval Estimation	769	37.13	Discussion	789
37.8	Bayesian Regression	769	37.13.1	Pitfalls in the Choice of Prior	789
37.8.1	Bayesian Linear Regression	769	37.13.2	Computationally Intensive	789
37.8.2	Hierarchical Bayesian Linear Regression	770	37.13.3	Bayesian Marginalisation Paradox	789
37.8.3	Generalised Linear Models	771	37.13.4	Bayesian Discrepant Posterior Phenomenon	790
37.9	Bayesian Computation	772	37.13.5	Lack of Inference Results	790
37.9.1	Laplace Approximation	772	References		791
37.9.2	The E-M Algorithm	773			
37.9.3	Monte Carlo Sampling	773			
37.9.4	Importance Sampling	774			
37.9.5	Rejection Sampling	774			
37.9.6	Markov Chain Monte Carlo	774			
37.10	Monitoring Convergence	780			
37.10.1	Geweke Diagnostic	780			
37.10.2	The Serial Correlation and the Effective Sample Size (ESS)	780			
37.10.3	Gelman-Rubin Diagnostic	780			
37.10.4	Model Robustness and Sensitivity Analysis	781			
37.10.5	Posterior Predictive Checking	781			

Abstract

Bayesian modelling has come a long way from the first appearance of the Bayes theorem. Now it is being applied in almost every scientific field. Scientists and practitioners are choosing to use Bayesian methodologies over the classical frequentist framework because of its rigour mathematical framework and the ability to combine prior information to define a prior distribution on the possible values of the unknown parameter. Here in this chapter we briefly discuss various aspects of Bayesian modelling. Starting from a short introduction on conditional probability, the Bayes theorem, different types of prior distributions, hierarchical and empirical Bayes and point and interval estimation, we describe Bayesian regression modelling with more detail. Then we mention an array of Bayesian computational techniques, viz. Laplace approximations, E-M

A. K. Chakraborty (✉) · S. Dey · P. Chakraborty · A. Chanda
 Indian Statistical Institute, Kolkata, West Bengal, India
 e-mail: akchakraborty123@rediffmail.com;
 soumenstat89@gmail.com; chakraborty.poulami94@gmail.com;
 aleenachanda03@gmail.com

algorithm, Monte Carlo sampling, importance sampling, Markov chain Monte Carlo algorithms, Gibbs sampler and Metropolis-Hastings algorithm. We also discuss model selection tools (e.g. DIC, WAIC, cross-validation, Bayes factor, etc.) and convergence diagnostics of the MCMC algorithm (e.g. Geweke diagnostics, effective sample size, Gelman-Rubin diagnostic, etc.). We end the chapter with some applications of Bayesian modelling and discuss some of the drawbacks in using Bayesian modelling in practice.

Keywords

Bayesian modelling · Prior distribution · Bayesian regression · Bayesian computation · Markov chain Monte Carlo · Gibbs sampler · Metropolis-Hastings algorithm · Bayesian model selection · WAIC · Cross-validation · Bayes factor

37.1 Introduction

Bayesian has a wide range of applications in various fields, namely: (a) medicine, (b) document classification, (c) information retrieval, (d) semantic search, (e) image processing, (f) spam filter and (g) reliability.

An advantage of Bayesian statistics is its ability to combine prior information regarding a parameter with the data and compute the posterior distribution. Thus, the prominent features of available information are used in Bayesian statistics to obtain the posterior distribution which can further be used as prior with the availability of new observations.

It is usually known that the larger the data, the better is the inference regarding the population. But there is no concept of ‘too little data’ in Bayesian statistics [1]. Bayesian method is helpful in modelling data with small sample sizes with a good prediction accuracy.

Bayesian statistics follows the ‘Likelihood Principle’ which states that all information from the data that is relevant to inferences about the value of the model parameters is in the equivalence class to which the likelihood function belongs.

The Bayesian concept of a credible interval is sometimes put forward as a more practical concept than the confidence interval.

Bayesian modelling framework provides a convenient setting for a wide range of models, such as hierarchical models and missing data problems. Numerical methods (e.g., Markov chain Monte Carlo, the Laplace approximation) allow computational tractability of these models.

Different methods such as importance sampling, rejection sampling, uniform sampling, Monte Carlo Markov chain and Gibbs sampling help to give reliable estimates of parameters involving tractable computations. Also, meaningful

inferences regarding parameters can be made in presence of missing observations.

37.2 Conditional Probability

Let n_1 denote the number of persons who smoke and have cancer. Let n_2 denote the number of persons who smoke. Let n denote the total number of persons. You are now asked to find the probability that a person will have cancer given that he smokes. To answer such questions, we have to know what conditional probability is. Conditional probabilities arise naturally in the investigation of experiments where an outcome of a trial may affect the outcomes of the subsequent trials. The conditional probability of an event B given an event A is the probability that the event B will occur given the knowledge that the event A has already occurred. This can be denoted by $P(B|A)$.

Bayes’ Theorem

In probability theory and statistics, Bayes’ theorem describes the probability of an event, based on prior knowledge of conditions that might be related to the event. For example, if cancer is related to age, then, using Bayes’ theorem, a person’s age can be used to more accurately assess the probability that they have cancer, compared to the assessment of the probability of cancer made without knowledge of the person’s age. Mathematically, Bayes’ theorem can be stated as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)},$$

where A and B are events and $P(B) \neq 0$. $P(A|B)$ is a conditional probability: the likelihood of event A occurring given that B is true. $P(B|A)$ is also a conditional probability: the likelihood of event B occurring given that A is true. $P(A)$ and $P(B)$ are the probabilities of observing A and B independently of each other; this is known as the marginal probability. In the Bayesian interpretation, probability measures a ‘degree of belief.’ Bayes’ theorem then links the degree of belief in a proposition before and after accounting for evidence, and hence the probabilities of Bayes’ theorem have the following interpretation: $P(A)$ is termed the prior, i.e. the initial degree of belief in A; $P(A|B)$ is termed the posterior, i.e. the degree of belief having accounted for B. The ratio $\frac{P(B|A)}{P(B)}$ represents the support B provides for A.

37.3 Some Examples

37.3.1 Prosecutor’s Fallacy

Prosecutor’s fallacy is a very famous application of Bayes’ rule. A fallacy for statistical reasoning, prosecutor’s fallacy, is used by the prosecution in the criminal trial to argue if

the person suspected is guilty. The problem is as follows: A number of persons in a city having a population of, say, 100, have been suspected of committing a crime, and one of the defendants is brought to trial. Further, the blood that has been found in the crime scene matches with that of the defendants'. The court has to judge if he/she is guilty or innocent. To solve the problem and to understand where the fallacy lies, let us first understand the following notations: G denotes the defendant is guilty, G^c denotes the defendant is innocent. B : Blood found in the crime scene is consistent with the defendant's blood. Let us understand the following conditional probabilities: $P(B|G^c)$ = Probability that a person who is not guilty matches the blood. $P(G^c|B)$ = Probability that a person who matches the blood description is innocent. Now the prosecutor's fallacy is, if $P(B|G^c) = 0.01$, then $P(G^c|B) = 0.01$. This means that if there is a 1% chance of a random person's blood type matching with that of the crime scene, then there is a 1% chance of the defendant being not guilty. Following the prosecutor's arguments, the conditional probabilities might seem similar. But the result of the conditional probability $P(G^c|B)$ using Bayes' theorem is inconsistent with the prosecutor's argument. Applying Bayes' theorem, $P(G|B) = \frac{P(B|G)P(G)}{P(B|G)P(G)+P(B|G^c)P(G^c)}$. Now if a person is guilty, then it is obvious that his/her blood type would match that at the crime scene. Thus, $P(B|G) = 1$. We also have $P(G|B) = \frac{1 \times P(G)}{[1 \times P(G)] + [0.01 \times P(G^c)]} = \frac{1}{0.99 + 0.01/P(G)}$.

37.3.2 Monty Hall Problem

The Monty Hall problem, a brain teaser, is a statistical puzzle named after its host Monty Hall, which was aired in the television show 'Let's Make a Deal'. In the show, each contestant was given an opportunity of winning the grand prize by selecting one of the three doors. The grand prize was a car which was placed behind one of the doors and the other two contained goats. After the contestant picked a door, Monte used to show him one of the two remaining doors which did not contain the car and the contestant was given an option to alter his decision. According to some, swapping would not be advantageous since there is a 50% chance that the car is behind one of the two unopened doors. However it can be shown using Bayes' theorem that swapping would lead to an approximate 66% chance of winning the car, whereas not swapping would lead to an approximate 33% chance of not winning the car. Let P denote the door containing the car, C denote the initial choice of the contestant and M denote the door shown by Monty. The car is placed behind a door randomly, prior to the selection made by the contestant. Thus, selection of the door is independent of the prize being placed behind a door, i.e. P is independent of C . Therefore, $P(P = p) = P(P = p|C = c) = 1/3, p = 1, 2, 3, c = 1, 2, 3$. However the door Monte shows depends on where the car is with the conditional probability, $P(M = m|P = p, C = c)$,

$c = 1, 2, 3, p = 1, 2, 3, m = 1, 2, 3$. WLG, we assume that the contestant selects door number 3. Define $Pr(P = 1) = p_1; Pr(P = 2) = p_2; Pr(P = 3) = p_3$; and $Pr(M = m|P = p, C = c) = a_{mpc}; m = 1, 2, 3; p = 1, 2, 3; c = 1, 2, 3$

$$P(P = 1|M = 2) = \frac{a_{213}p_1}{a_{213}p_1 + a_{223}p_2 + a_{233}p_3}$$

Since the contestant selects door number 3, Monty would either show door number 1 or 2 depending on where the car is. Let us say, if the car is behind door number 2, then Monty would show door number 1 and vice versa. However if the car is behind door number 3, then Monty can show door number 1 or 2. The possible cases are as follows:

m	1	2	3
$P(M=m P=1,C=3)$	0	1	0
$P(M=m P=2,C=3)$	1	0	0
$P(M=m P=3,C=3)$	0.5	0.5	0

Therefore, $P(P=1 | M=2) = \frac{1 \times 1/3}{[1 \times 1/3] + [0 \times 1/3] + [0.5 \times 1/3]} = \frac{2}{3}$.
 Again, $P(P = 3|M = 2) = \frac{a_{233}p_3}{a_{213}p_1 + a_{223}p_2 + a_{233}p_3} = \frac{0.5 \times 1/3}{[1 \times 1/3] + [0 \times 1/3] + [0.5 \times 1/3]} = \frac{1}{3}$. It is obvious that $P(P = 2|M = 2) = 0$, since Monty would not reveal the door containing the car. Thus, it can be seen that if the contestant alters their decision, then the chances of getting the car are higher.

37.3.3 Types of Prior

Improper Prior

If the prior distribution is not previously known, it is determined on a subjective or theoretical basis, which may lead to the situation $\int_{\Theta} \pi(\theta)d\theta = \infty$. In such cases the prior distribution is said to be improper (or generalised). But as long as the posterior distribution is defined, one can use the Bayesian method, even if the prior is improper.

Example

Let us consider a pdf $f(x|\theta) = N(\theta, 1)$ and $\pi(\theta) = c$, where c is an arbitrary constant. The posterior distribution is given by: $p(\theta|y) = \frac{f(x|\theta)p(\theta)}{\int_{-\infty}^{\infty} f(x|\theta)p(\theta)d\theta} = \frac{c \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(y-\theta)^2}}{\int_{-\infty}^{\infty} c \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(y-\theta)^2} d\theta}$. Thus, $\theta|y \sim N(y, 1)$. Hence, the prior is improper, but the posterior is tractable. ◀

Non-informative Prior

Non-informative priors favour no particular values of the parameter over others. If Θ is finite, we choose uniform prior. If Θ is infinite, there are several ways of choosing a prior distribution some of which may be improper.

Location Density For a location density $f(x|\theta) = f(x-\theta)$, the location-invariant non-informative prior is $\pi(\theta) = \pi(0)$. (a constant). We usually choose $\pi(0) = 1$ (improper prior).

Example

Suppose $X \sim N(\theta, \sigma^2)$. Let $Y = X + c$. Then $Y \sim N(\eta, \sigma^2)$, where $\eta = \theta + c$. Thus, Y has the density $f_y(y) = f(y-\eta) = \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-(y-\eta)^2}{2\sigma^2}$. Assume $\pi(\theta)$ and $\pi^*(\eta)$ are the non-informative priors of θ and $\eta = \theta+c$, respectively. Let A be any set. Then, $\int_A \pi(\theta)d\theta = \int_A \pi^*(\eta)d\eta = \int_{A-c} \pi(\theta)d\theta$ (because $\eta \in A \Leftrightarrow \theta + c \in A \Leftrightarrow \theta \in A-c$) $= \int_A \pi(\theta - c)d\theta$. Thus, $\pi(\theta) = \pi(\theta - c)$. Now, let $\theta = c$. Then $\pi(c) = \pi(0)$, a constant. It is generally convenient to choose $\pi(c) = 1$. ◀

Scale Density

For a scale density, $f(x|\sigma) = \frac{1}{\sigma} f(\frac{x}{\sigma})$, for $\sigma > 0$, the scale-invariant non-informative prior is $\pi(\sigma) \propto \frac{1}{\sigma}$.

Example

Let $X \sim N(0, \sigma^2)$ and $Y = cX$. Then, $Y \sim N(0, c^2\sigma^2)$. The density of Y is $f_y(y) = \frac{1}{\sqrt{2\pi}c\sigma} \exp -\frac{y^2}{2c^2\sigma^2} = \frac{1}{\sqrt{2\pi}\eta} \exp -\frac{y^2}{2\eta^2} = \frac{1}{\eta} f(\frac{y}{\eta})$, where $\eta = c\sigma$. ◀

Assume $\pi(\theta)$ and $\pi^*(\eta)$ are the non-informative priors of σ and $\eta = c\sigma$, respectively. Let A be any set. Then, $\int_A \pi(\sigma)d\sigma = \int_A \pi^*(\eta)d\eta = \int_{A/c} \pi(\sigma)d\sigma$ (because $\eta \in A \Leftrightarrow c\sigma \in A \Leftrightarrow \sigma \in A/c$) $= \int_A \pi(\frac{\sigma}{c}) \frac{1}{c} d\sigma$.

Thus, $\pi(\sigma) = \pi(\frac{\sigma}{c}) \frac{1}{c}$. Let, $\sigma = c$. Then, $\pi(c) = \pi(1) \frac{1}{c}$.

It is generally convenient to set $\pi(1)=1$. Then, $\pi(c) = \frac{1}{c} \Leftrightarrow$

$\pi(\sigma) = \frac{1}{\sigma}$. Note that, $\pi(\sigma) = \frac{1}{\sigma}$ is an improper prior since $\int_0^\infty \frac{1}{\sigma} d\sigma = \infty$. Jeffreys' prior is given by: $\pi(\theta) \propto I(\theta)^{1/2}$.

Example

$f(x|p) = \binom{n}{x} p^x(1 - p)^{n-x}$. $I(p) = \frac{n}{p(1-p)}$ Let y_1, y_2, \dots, y_n be n independent samples from $N(\beta x_i, 1)$ distribution where β is an unknown parameter and x_1, x_2, \dots, x_n are known. Then, $I(\beta) = S_{xx}$, where $S_{xx} = \sum_{i=1}^n x_i^2$. So Jeffreys' prior is proportional to $S_{xx}^{\frac{1}{2}}$. The posterior distribution has a $N(\frac{S_{xy}}{S_{xx}}, S_{xx}^{-1})$ where $S_{xy} = \sum_{i=1}^n x_i y_i$ ◀

Conjugate Prior

A family F of probability distributions on Θ is said to be conjugate or closed under sampling for a likelihood $f(x|\theta)$,

if for every prior $\pi \in F$, the posterior distribution $\pi(\theta | x)$ also belongs to F . Choosing a conjugate prior makes computation of posterior quantities easier. Further, updating the distribution under the new data is also convenient.

Example

Let $y \sim Poisson(\theta)$, $\theta \in (0, \infty) = \Theta$. $\theta \sim Gamma(c, \alpha)$. Then $\theta|y \sim \Gamma(c + 1, \alpha + y)$. $y|\theta \sim Bin(n, \theta)$ for $y = 0, 1, 2, \dots, n$. $f(y|\theta) = \binom{n}{y} \theta^y (1 - \theta)^{n-y}$. $\theta \sim Beta(a, b)$ with density $p(\theta) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \theta^{a-1} (1-\theta)^{b-1}$. The posterior distribution of $\theta|y$ is given by: $\theta|y \sim Beta(y + a, n - y + b)$. The posterior distribution can be obtained by applying Bayes' theorem to the densities. $p(\theta|y) = \frac{f(y|\theta)p(\theta)}{\int f(y|\theta)p(\theta)d\theta} = \frac{\Gamma(n+a+b)}{\Gamma(y+a)\Gamma(n-y+b)} \theta^{y+a-1} (1-\theta)^{n-y+b-1}$.

Now $y_1, y_2, \dots, y_n | \theta \sim N(\theta, \sigma^2)$. Let $\tau = \frac{1}{\sigma^2}$. The likelihood function is given by: $L(y|\theta) = \prod_{i=1}^n f(y_i|\theta) = (\tau/2\pi)^{n/2} e^{-\tau/2 \sum_{i=1}^n (y_i - \theta)^2}$. $\theta \sim N(\theta_0, 1/\tau_0)$. Hence, the posterior distribution is given by: $\theta|y \sim N(\frac{\tau_0}{\tau_0+n\tau} \theta_0 + \frac{n\tau}{\tau_0+n\tau} \bar{y}, \frac{1}{\tau_0+n\tau})$. ◀

The Multinomial Distribution and the Dirichlet Prior

Suppose an experiment is conducted having n independent trials and each trial produces exactly one of the k events E_1, E_2, \dots, E_k with respective probabilities $\theta_1, \theta_2, \dots, \theta_k$. These k events are mutually exclusive and collectively exhaustive. Here, $\theta_1 + \theta_2 + \dots + \theta_k = 1$. Define:

- $X_1 =$ number of trials in which E_1 occurs.
- $X_2 =$ number of trials in which E_2 occurs.
- \vdots
- $X_k =$ number of trials in which E_k occurs.

Then $X = (X_1, X_2, \dots, X_k)$ is said to have a multinomial distribution with index n and parameter $\pi = (\theta_1, \theta_2, \dots, \theta_k)$. In most problems, n is regarded as fixed and known. Then, $f(X|\pi) \propto \prod_{i=1}^k \theta_i^{X_i}$. We here introduce the Dirichlet distribution which is nothing but the multivariate generalisation of the beta distribution as the conjugate prior distribution for the multinomial distribution. The pdf is given by: $\pi(\theta_1, \theta_2, \dots, \theta_k | \alpha_1, \alpha_2, \dots, \alpha_k) \propto \prod_{i=1}^k \theta_i^{\alpha_i - 1}$. We say that $(\theta_1, \theta_2, \dots, \theta_k) \sim Dirichlet(k; (\alpha_1, \alpha_2, \dots, \alpha_k))$. The posterior distribution of $\theta_1, \theta_2, \dots, \theta_k$ given X also follows Dirichlet with parameter set $(k; \alpha_1 + X_1, \alpha_2 + X_2, \dots, \alpha_k + X_k)$.

37.3.4 Reference Prior

A reference prior distinguishes between the parameters of interest and the nuisance parameters. Let $X \sim f(x|\theta)$

where $\theta = (\theta_1, \theta_2, \dots, \theta_p, \dots, \theta_q)$. Also, define $\theta_A = (\theta_1, \theta_2, \dots, \theta_q)$ and $\theta_B = (\theta_{q+1}, \dots, \theta_p)$. Let $\pi(\theta_B|\theta_A)$ be Jeffreys' prior associated with $f(x|\theta)$, when θ_A is fixed. To obtain the reference prior, we then derive the marginal distribution $f^*(x|\theta_A) = f^*(x|\theta_1, \theta_2, \dots, \theta_q) = \int \int \dots \int f(x|\theta) \pi(\theta_{q+1}, \dots, \theta_p|\theta_1, \theta_2, \dots, \theta_q) d\theta_{q+1} \dots d\theta_p$ and then compute Jeffreys' prior associated with $f^*(x|\theta_A)$. In this process, the nuisance parameter gets eliminated by using a Jeffreys' prior where the parameters of interest remain fixed.

Example

Let us consider the Neyman-Scott problem. Here, $x_{ij} \sim N(\mu_i, \sigma^2), i = 1, 2, \dots, n; j = 1, 2$. The goal is to estimate the variance of the whole population σ^2 . After several steps of calculation, the usual Jeffreys' prior is obtained as: $\pi(\mu_1, \mu_2, \dots, \mu_n, \sigma) = \sigma^{-n-1}$. But, by action $\mathbb{E}[\sigma^2|x_{11}, \dots, x_{n2}] = s^2/(n-2); s^2 = \sum_{i=1}^n \frac{(x_{i1}-x_{i2})^2}{2}$. The posterior expectation converges to $\sigma^2/2$. In our example $\theta_A = \sigma$ and $\theta_B = (\mu_1, \mu_2, \dots, \mu_n)$. The reference prior associated with θ_A and θ_B gives a flat prior for θ_B ; θ_B being a location parameter. Then, $f^*(x|\theta_A) = \prod_{i=1}^n e^{-(x_{i1}-x_{i2})^2/4\sigma^2} \frac{1}{\sqrt{2\pi}2\sigma}$ is a scale family and $\pi(\sigma) = \frac{1}{\sigma}$. Therefore, by action $\mathbb{E}[\sigma^2|x_{11}, \dots, x_{n2}] = s^2/(n-2)$ which is consistent for σ . ◀

37.4 Hierarchical Bayes

The choice of prior in Bayesian is subjective. So, the subjective part should be modified. On the other hand, the prior information is barely rich enough to define the prior distribution exactly. So, it is necessary to include this uncertainty in the model. Under hierarchical Bayes' analysis, the prior information decomposes the prior distribution into several parts. It improves the robustness of the resulting Bayes' estimate.

A Bayesian statistical model is hierarchical if it involves several levels of conditional prior distributions. This particular modelling of the prior information decomposes the prior distribution into several conditional levels of distributions so as to introduce a distinction between structural and subjective parts of information. In the simplest case, the hierarchical structure is used to two levels, the parameters of the first being associated with a prior distribution defined on the second. The first-level distribution is generally a conjugate prior for computational tractability.

One noteworthy drawback of the hierarchical Bayes method is that often it is difficult to derive the explicit Bayes' estimate even when the successive levels are conjugate. In such cases, one has to depend on numerical computations for obtaining posterior estimates.

Example 1 Software is to be tested before release in the market so that it carries minimum number of errors while using it. In a software testing problem, a success occurs when the software fails to provide the right output for an input. The size of a bug is defined as the number of inputs that would have eventually passed through the bug, if the bug were not fixed. From the definition of the size of the bug, the appearance of a bug may be termed as a success in a sequence of independent trials. Hence, we say that $S_i|t_i \sim Bin(n_i, t_i)$ where $S_i =$ Eventual size of the i^{th} bug; $i = 1, 2, \dots, m$. Now there should be a distribution for the probability of detection of a bug t_i , since it would vary for different bugs. One reasonable distribution is that t_i 's are i.i.d. from a beta distribution, say, $t_i|\alpha_1, \alpha_2 \sim Beta(\alpha_1, \alpha_2)$ as it is a conjugate prior. Hyperpriors can also be assigned on the parameters α_1, α_2 by deducing from domain knowledge on quantities that are comparatively easier to obtain, e.g., mean and variance.

Example 2 The regression set-up is as follows: $y_{ij} = \alpha + \beta_i + e_{ij}$ where β_i 's are identically and independently distributed as $N(0, \frac{1}{\gamma_1})$; e_{ij} 's are identically and independently distributed as $N(0, \frac{1}{\gamma_2}); i = 1, 2, \dots, p; j = 1, 2, \dots, n_i$. Here, β_i and e_{ij} 's are all independent. Thus, Y follows a multivariate normal distribution with y_{ij} following $N(\alpha, \frac{1}{\gamma_1} + \frac{1}{\gamma_2})$, where

$$Cov(y_{ij}, y_{i'j'}) = \begin{cases} 0 & \text{if } i \neq i' \\ 1/\gamma_1 & \text{if } i = i', j \neq j' \end{cases}$$

However, in Bayesian approach, the traditional method of writing the sampling model takes altogether a different notation: $y_{ij}|\eta_i, \gamma_1$ independently follows $N(\eta_i, \frac{1}{\gamma_1})$, and $\eta_i|\alpha, \gamma_1$ independently follows $N(\alpha, \frac{1}{\gamma_1})$. Here, we can see that $y_{ij} \perp \alpha, \gamma_1|\eta_i, \gamma_2$ and $\eta_i \perp \gamma_2|\alpha, \gamma_1$. It can be easily shown that: $y_{ij}|\eta_i, \gamma_1, \gamma_2$ independently follows $N(\eta_i, \frac{1}{\alpha_2}); \eta_i|\alpha, \gamma_1, \gamma_2$ independently and identically distributed as $N(\alpha, \frac{1}{\gamma_1}); i = 1, 2, \dots, p$. Since $\alpha, \gamma_1, \gamma_2$ are not known, we can specify normal, gamma and gamma priors for them respectively.

37.5 Empirical Bayes

In this method, we approximate the prior distribution by frequentist methods when the prior distribution is vague. This method can be considered as a dual method of hierarchical Bayes' analysis. It is asymptotically equivalent to

the Bayesian approach. Further, it may be an acceptable approximation in problems for which a proper Bayesian modelling is too complicated.

37.5.1 Non-parametric Empirical Bayes

Given $n+1$ independent observations x_1, x_2, \dots, x_{n+1} , with densities $f(x_i|\theta_i)$, the problem is to draw an inference on θ_{i+1} under the additional assumption that θ_i 's have all been generated according to some unknown prior distribution g , i.e. the sampling distribution $f(x_i|\theta_i)$ is known but the prior distribution is unknown.

The marginal distribution $f_g(x) = \int_{\Theta} f(x|\theta)g(\theta)d\theta$. This marginal distribution can be used to recover the distribution g from the observations, since x_1, x_2, \dots, x_n can be considered as an i.i.d. sample from $f_g(x)$. Deriving an approximation \tilde{g}_n in this way, we can use it as a substitute for the two prior distributions and prepare the approximation to the posterior distribution as

$$\tilde{\pi}(\theta_{n+1}|x_{n+1}) \propto f(x_{n+1}|\theta_{n+1})\tilde{g}_n(\theta_{n+1}).$$

This approach is not Bayesian. In Bayesian, ignorance on g would index this distribution by a hyperparameter λ and second level prior distribution $\pi_2(\lambda)$.

37.5.2 Parametric Empirical Bayes

In an exponential family setting, a sample prior choice for $f(x|\theta)$ is the conjugate prior, say $\pi(\theta|\lambda)$. The Bayesian school models uncertainty by a probability distribution over parameters. The choice of prior distribution is the most important criterion, since choosing an alternate prior could lead to a more robust inference. Let us define θ as the unknown state of nature, i.e. the parameter, and X as the data that provides information about θ . Suppose $X \sim f(x|\theta)$, where f is the possible distribution of X . In addition to having a model $f(x|\theta)$ for the data X , Bayesian assumes a distribution for θ . This distribution is called the prior distribution or simply a prior, since it quantifies the uncertainty about the parameter prior to observing the data. If the prior represents Bayesian's subjective belief and knowledge, then it is called a subjective prior. On the other hand, if it is a conventional representing small or no information, then it is an objective prior.

Given the model for the data and the prior $\pi(\theta)$ of θ , the conditional distribution of θ given $X = x$ is given by Bayes' formula, $\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int f(x|\theta)\pi(\theta)d\theta}$. To get a composite picture about final belief about θ , the posterior distribution comes into the scenario which combines the prior belief about the parameter θ and also any available sample information

about θ . Thus we can very easily say that the posterior distribution reflects the updated belief about θ after the data is being observed since the transition from the prior to the posterior is basically what we have learnt from the data. If the posterior distribution can be known, instead of reporting the whole posterior distribution, one may also report the summary descriptive measures associated with it like mean, variance, median and others. One may also use the posterior distribution to answer questions related to more structural problems like estimation and testing.

37.6 Point Estimation

37.6.1 Univariate Case

A point estimate of θ , say, $\hat{\theta}$, is obtained by simply using the commonly used summaries of the posterior distribution like mean, median and mode. Again, when we consider a flat prior for θ , the maximum likelihood estimate can also be used as an estimate of the parameter θ since here MLE simply becomes the mode. However, when the posterior distribution is symmetric, one may report the mean or the median as they become equal. If, in particular, this distribution is unimodal, the value of mode will also be equal to that of the mean and the median. A measure of extent of uncertainty is also reported together with the point estimate. We can look at the posterior SD or variance together with the posterior mean to obtain a measure of accuracy of the point estimate, $\theta(\hat{y})$. The posterior variance is given by:

$$\mathbb{E}_{\theta|y}[\theta - \theta(\hat{y})]^2.$$

Let the posterior mean be given by $\mathbb{E}_{\theta|y}(\theta) = \mu_y$; we may write

$$\begin{aligned} & \mathbb{E}_{\theta|y}[\theta - \theta(\hat{y})]^2 \\ &= \mathbb{E}_{\theta|y}[\theta - \mu + \mu - \theta(\hat{y})]^2 \\ &= \text{Var}_{\theta|y}[\theta] + (\mu - \theta(\hat{y}))^2. \end{aligned}$$

Thus, we can very easily see that if $\theta(\hat{y}) = \mu(y)$, then the posterior variance is minimised with respect to $\theta(\hat{y})$ over all point estimators $\theta(\hat{y})$. Our aim is to define the optimum estimate. So, we bring in the concept of loss function.

Define $L(\theta, a)$ as the cost of estimating the parameter θ by action a . Average loss in estimating the parameter θ by action a is given by $\mathbb{E}_{\theta|y}[L(\theta|a)] = \int_{\Theta} L(\theta, a)\pi(\theta|y)d\theta$. The estimate of θ where this loss is minimised is the best estimate called Bayes' estimate.

37.6.2 Multivariate Case

We denote estimate of a vector of parameters θ as $\hat{\theta} = (\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_p)$. The posterior mode and posterior median can be considered as estimates of θ . However computation of these quantities can be difficult to carry out in practice. Another commonly used estimator of θ is its posterior mean $\mu = \mathbb{E}_{\theta|y}[\theta]$ because it is well defined and a measure of uncertainty can be captured by the posterior covariance matrix, $V = \mathbb{E}_{\theta|y}[(\theta - \mu)(\theta - \mu)']$. It can be shown that $\mathbb{E}_{\theta|y}[(\theta - \hat{\theta}(y))(\theta - \hat{\theta}(y))'] = V + (\mu - \hat{\theta}(y))(\mu - \hat{\theta}(y))'$. Thus the posterior mean minimizes the posterior covariance.

37.7 Bayesian Interval Estimation

The confidence interval in the Bayesian approach referred to as a ‘credible set’ is analogous to that of the confidence interval concept in the frequentist approach. A set $C \subset \Theta$ is called a $100(1 - \alpha)\%$ credible set of θ if $P(C|data) \geq 1 - \alpha$, where $P(C|data)$ is expressed as $\int_C p(\theta|y)d\theta$ (if θ is continuous) or $\sum_C p(\theta|y)$ (if θ is discrete). In other words, the probability that θ lies in C given the observed data is at least $(1 - \alpha)$, i.e. it gives direct probability statements about the likelihood of θ falling in C . However, the frequentist definition of confidence interval is that if C is computed for a large number of data sets, about $(1 - \alpha)100\%$ of them would contain the true value of θ . Suppose there is only one data set. Then according to the Bayesian definition, the credible set would either contain the credible set or not. In other words, it would give an actual coverage probability (1 or 0). However according to the frequentist definition, it won’t be possible to compute C , as there is only one data set. Then if θ belongs to the 90% confidence interval, then it would belong to the 95% confidence interval as well. Credible sets are not unique. So how to determine which set would be most suitable? A technique for doing this is given by the highest posterior density, or HPD credible set, defined as the set

$C = \{\theta \in \Theta : p(\theta|data) \geq k(\alpha)\}$, where $k(\alpha)$ is the largest constant satisfying $P(C|data) \geq 1 - \alpha$.

37.8 Bayesian Regression

Regression is a widely used method in statistics, so the Bayesian approach to it is also important to practitioners. There are two main goals for doing a regression analysis: (i) to understand how some predictor(s) influence the values of the variable of interest and (ii) to predict the value of the variable of interest for some new value of predictor variable. Here we shall discuss some basics of regression analysis under the Bayesian approach.

37.8.1 Bayesian Linear Regression

Regression studies the relationship between two or more variables. Let y denote the variable of interest, also called the *response variable*, and $\mathbf{x} = (x_1, \dots, x_r)'$ denotes a vector of *predictor variables* or *covariates* which may be used to explain y . Regression method can be viewed as a problem of predicting y using the predictor variables x_1, \dots, x_r . The distribution of y given \mathbf{x} is usually studied, and the conditional mean $h(\mathbf{x}) := \mathbb{E}(y|\mathbf{x})$ can be obtained as the best predictor with respect to mean squared error criterion [2]. However in practice, the exact expression of $h(\mathbf{x})$ is not known, so the function $h(\mathbf{X})$ is needed to be estimated using data. $h(\mathbf{x})$ can be modelled in different ways; the simplest and commonly used approach is to use a *linear regression model*

$$h(\mathbf{x}) = \beta_1 x_1 + \dots + \beta_r x_r = \mathbf{x}'\boldsymbol{\beta}, \quad (37.1)$$

where $\boldsymbol{\beta} = (\beta_1, \dots, \beta_r)'$ is vector of unknown regression parameters.

Next we describe Bayesian regression with a normal linear model. Suppose we have n observations on y and also on each of the r components of \mathbf{x} : $\{(y_i, \mathbf{x}_i') = (y_i, x_{i1}, \dots, x_{ir})' : i = 1, \dots, n\}$. In matrix notation we write $\mathbf{y} = (y_1, \dots, y_n)'$, $\mathbf{X} = ((x_{ij}))_{1 \leq i \leq n, 1 \leq j \leq r}$. We assume the observation errors are independently and identically distributed normal variable with mean zero and unknown variance σ^2 . This yields the linear model

$$\mathbf{y} | \mathbf{X}, \boldsymbol{\beta}, \sigma^2 \sim \mathcal{N}_n(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_n), \quad (37.2)$$

where \mathbf{I}_n denotes the $n \times n$ identity matrix. Then the parameter vector is $(\boldsymbol{\beta}, \sigma^2)$. The Bayesian approach to inference now requires a prior distribution on $(\boldsymbol{\beta}, \sigma^2)$. A standard prior distribution for this normal regression model assumes a non-informative uniform prior on $(\boldsymbol{\beta}, \log \sigma)$, or equivalently, the prior density on $(\boldsymbol{\beta}, \sigma^2)$,

$$\pi(\boldsymbol{\beta}, \sigma^2) \propto \sigma^{-2}. \quad (37.3)$$

For a large sample size and a relatively smaller dimensional $\boldsymbol{\beta}$, this non-informative prior distribution gives acceptable results and takes less effort than specifying prior knowledge in probabilistic form. We factor the joint posterior distribution of $\boldsymbol{\beta}$ and σ^2 as

$$\pi(\boldsymbol{\beta}, \sigma^2 | \mathbf{y}) = \pi_1(\boldsymbol{\beta} | \sigma^2, \mathbf{y}) \pi_2(\sigma^2 | \mathbf{y}). \quad (37.4)$$

Conditional posterior distribution of $\boldsymbol{\beta}$ given σ^2 can be obtained as

$$\boldsymbol{\beta} | \sigma^2, \mathbf{y} \sim \mathcal{N}_r(\hat{\boldsymbol{\beta}}, \sigma^2 (\mathbf{X}'\mathbf{X})^{-1}), \quad (37.5)$$

where $\hat{\beta} = (X'X)^{-1}X'y$. The marginal posterior distribution of σ^2 can be obtained as

$$\sigma^2 | y \sim \text{Inv-}\chi^2(n - r, s^2), \tag{37.6}$$

where $s^2 = (y - X\hat{\beta})'(y - X\hat{\beta}) / (n - r)$. The marginal posterior density of β , i.e. $\beta | y$, has a multivariate- t distribution with degrees of freedom $(n - r)$, mean $\hat{\beta}$ and shape matrix $s^2(X'X)^{-1}$, provided $(n - r) > 2$. The $100(1 - \alpha)\%$ highest posterior density (HPD) credible region for β is given by the ellipsoid

$$\{\beta : (\beta - \hat{\beta})'X'X(\beta - \hat{\beta}) \leq rs^2F_{r,n-r}(\alpha)\}, \tag{37.7}$$

where $F_{r,n-r}(\alpha)$ is the $(1 - \alpha)$ quantile of the $F_{r,n-r}$ distribution. For any analysis based on an improper prior distribution, it is important to check the propriety of the posterior distribution. The posterior $\pi(\beta, \sigma^2 | y)$ is proper if $n > r$ and rank of X is of full column rank r .

Suppose we wish to predict a new data set \tilde{y} with observed predictor variables \tilde{X} . The *posterior predictive density* of \tilde{y} is obtained as

$$f(\tilde{y} | y) = \int f(\tilde{y} | \beta, \sigma^2) \pi(\beta, \sigma^2 | y) d\beta d\sigma^2. \tag{37.8}$$

Note that the posterior predictive distribution has two sources of variation: the first source is due to the observational variability in the model and the second source of variability is coming from the joint posterior distribution of the parameters β and σ^2 . The above integration can be performed in two steps: first by averaging over the conditional posterior density $\pi_1(\beta | \sigma^2, y)$, we obtain the conditional posterior predictive density $f(\tilde{y} | \sigma^2, y)$, which is distributed as a normal variable with mean $\tilde{X}\hat{\beta}$ and covariance matrix $\sigma^2(I_n + \tilde{X}(X'X)^{-1}\tilde{X}')$. In the second step, we average this conditional posterior predictive density $f(\tilde{y} | \sigma^2, y)$ over the marginal posterior distribution $\pi_2(\sigma^2 | y)$. The resulting posterior predictive distribution $f(\tilde{y} | y)$ is the multivariate- t with mean $\tilde{X}\hat{\beta}$, shape matrix $s^2(I_n + \tilde{X}(X'X)^{-1}\tilde{X}')$ and $(n - r)$ degrees of freedom. To draw a random sample \tilde{y} from the posterior predictive distribution, we first draw (β, σ^2) from their joint posterior distribution and then draw a new \tilde{y} from $f(\tilde{y} | \beta, \sigma^2)$. Bayesian regression with unequal variances or unknown covariance matrix follows a similar path as discussed in [3].

Bayesian regression with unequal variances or an $n \times n$ covariance matrix follows a similar path as discussed above. If we replace the covariance matrix $\sigma^2 I_n$ with a symmetric positive definite matrix Σ in (37.2), the model becomes

$$y | X, \beta, \Sigma \sim \mathcal{N}_n(X\beta, \Sigma). \tag{37.9}$$

Assuming a non-informative uniform prior for β (i.e. $\pi(\beta | \Sigma) \propto 1$), the conditional posterior distribution of β given y and Σ follows a multivariate normal distribution with mean $\hat{\beta} = (X'\Sigma^{-1}X)^{-1}X'\Sigma^{-1}y$ and covariance matrix $(X'\Sigma^{-1}X)^{-1}$. The matrix inversion to compute the posterior covariance can be made computationally efficient by first computing a Cholesky factor of Σ , viz. $\Sigma^{1/2}$, and then by obtaining the upper triangular matrix R of a QR decomposition of $\Sigma^{-1/2}X$. Note that $(X'\Sigma^{-1}X)^{-1} = R^{-1}(R^{-1})'$ and computing inverse of an upper triangular matrix R is comparatively easier. For known Σ , we can carry out the posterior inference on β as described above. However unknown Σ , we need to obtain the posterior distribution of Σ as well. Assuming $\pi(\Sigma)$ denotes the prior density of Σ , the posterior density of Σ is then obtained from the following identity:

$$\pi(\Sigma | y) = \frac{\pi(\beta, \Sigma | y)}{\pi(\beta | \Sigma, y)} \propto \frac{f(y | X, \beta, \Sigma)\pi(\beta | \Sigma)\pi(\Sigma)}{\pi(\beta | \Sigma, y)}, \tag{37.10}$$

which must hold for any β . This calculation in (37.10) often gets very complex and depends on the prior density $\pi(\Sigma)$, and giving a prior for Σ is not an easy task. The practitioners often consider re-parameterisation of Σ to simplify the complexity, e.g. $\Sigma = \sigma^2\Sigma_0$, where Σ_0 is known, but σ^2 is unknown.

37.8.2 Hierarchical Bayesian Linear Regression

In practice, we often encounter data that are grouped into different hierarchical levels. For example, if we consider exam scores obtained by students as data, then it is organised at various hierarchical levels: students, class, school, district, etc. Analysis of hierarchical data should be analysed in an appropriate statistical framework that accounts for the different hierarchy, such as hierarchical linear modelling.

Hierarchical linear modelling can be thought of as an extension of conventional linear modelling where variability in the response variable is analysed with the predictor variables being at different hierarchical levels. Hierarchical linear models are sometimes called multi-level linear models or nested models. Hierarchical Bayesian linear model is essentially a hierarchical linear model that is estimated using Bayesian methods.

Let us explain the idea of hierarchical Bayesian linear modelling with an example. Assume that each y_j , $j = 1, 2, \dots, J$ is a vector of observation (length n_j) from a $\mathcal{N}_{n_j}(X_j\beta_j, \sigma_j^2 I_{n_j})$ population where the parameter $\beta_j = (\beta_{1j}, \dots, \beta_{rj})'$ governs the data generating process for y_j , $j = 1, 2, \dots, J$. We also assume that each of the parameters β_1, \dots, β_J follows a $\mathcal{N}_r(\alpha\mathbf{1}, \Lambda)$ and that they

are exchangeable, i.e. the joint distribution of β_1, \dots, β_j is invariant under permutations of the indices [4, 5]. The Bayesian hierarchical model contains the following stages:

$$\text{Stage 1 (Observation process): } y_j | \mathbf{X}_j, \beta_j, \sigma_j^2 \sim \mathcal{N}_{n_j}(\mathbf{X}_j \beta_j, \sigma_j^2 \mathbf{I}_{n_j})$$

$$\text{Stage 2 (Population process): } \beta_j | \alpha, \Lambda \sim \mathcal{N}_r(\alpha \mathbf{1}, \Lambda), \sigma_j^2 \sim \text{Inv-Gamma}(\tau_a, \tau_b)$$

$$\text{Stage 3 (Hyperprior distribution): } \alpha | \alpha_0, \sigma_\alpha^2 \sim \mathcal{N}(\alpha_0, \sigma_\alpha^2)$$

Here we assume Λ , τ_a , τ_b , α_0 and σ_α^2 as known quantity. However the above hierarchical Bayesian linear model can be extended to allow specification of prior distributions on these parameters if they are unknown to the practitioners. Obtaining posterior inference on such models is often not straightforward, and advance computing techniques are required to facilitate the inference. We shall discuss them in Sect. 37.9.

37.8.3 Generalised Linear Models

Generalised linear models are widely used in most areas of statistical applications. This class of modelling is applied in cases where the linear relationship between predictor variable \mathbf{x} and conditional mean of response variable $\mathbb{E}(y | \mathbf{x})$ is not appropriate (*cf.*, we have defined $\mathbb{E}(y | \mathbf{x}) = \mathbf{x}'\beta$ in the above examples of linear models). For example, in the cases where response y takes only the value 0 or 1 (binary), then a linear model will not be appropriate since the linear function $\mathbf{x}'\beta$ is likely to predict values other than 0 or 1 which are out of bounds for y . There are two components of a generalised linear model: (i) the conditional distribution of the response variable y given predictor \mathbf{x} (a matrix with dimension $n \times r$) with mean $\mathbb{E}(y | \mathbf{x}) = \mu$ and (ii) an invertible function $h(\cdot)$, called the 'link function, that relates the mean of the response variable to the linear predictor $\mathbf{x}'\beta$, i.e. $h(\mu) = \mathbf{x}'\beta$. Logistic regression, Poisson regression and probit regression are very popular examples of generalised linear models. We shall give examples of these popular regression models below.

Logistic Regression

Logistic regression is a very popular example of generalised linear modelling for binary data. Assume that $\mathbf{y} = (y_1, \dots, y_n)'$ is a vector of random variables where each component y_i independently follows a binomial distribution with parameters m_i and p_i . Note that $\mathbb{E}(y_i/m_i) = p_i$. Now we need to specify the link function to complete the generalised linear model. $\text{logit}(\cdot)$ is a very common choice for link function under this model setting: $h(p_i) = \text{logit}(p_i) = \log(\frac{p_i}{1-p_i}) = \mathbf{x}'_i \beta$.

The model likelihood function for β is

$$l(\beta | \mathbf{y}) = \prod_{i=1}^n \binom{m_i}{y_i} \left(\frac{\exp(\mathbf{x}'_i \beta)}{1 + \exp(\mathbf{x}'_i \beta)} \right)^{y_i} \left(1 - \frac{\exp(\mathbf{x}'_i \beta)}{1 + \exp(\mathbf{x}'_i \beta)} \right)^{m_i - y_i}. \quad (37.11)$$

Note that the logit link maps its argument with domain $(0, 1)$ to the whole real line, which is appropriate for a linear predictor. One then can fix a prior for the regression coefficient β to complete Bayesian formulation (e.g. a $\mathcal{N}_r(\alpha \mathbf{1}, \sigma_\beta^2 \mathbf{I}_r)$ prior or a non-informative uniform prior, etc.). Suppose $\pi(\beta)$ denotes the prior density; then the posterior density of β is

$$\pi(\beta | \mathbf{y}) = \frac{\pi(\beta) \prod_{i=1}^n \binom{m_i}{y_i} \left(\frac{\exp(\mathbf{x}'_i \beta)}{1 + \exp(\mathbf{x}'_i \beta)} \right)^{y_i} \left(1 - \frac{\exp(\mathbf{x}'_i \beta)}{1 + \exp(\mathbf{x}'_i \beta)} \right)^{m_i - y_i}}{\int_{\mathbf{b}} \pi(\mathbf{b}) \prod_{i=1}^n \binom{m_i}{y_i} \left(\frac{\exp(\mathbf{x}'_i \mathbf{b})}{1 + \exp(\mathbf{x}'_i \mathbf{b})} \right)^{y_i} \left(1 - \frac{\exp(\mathbf{x}'_i \mathbf{b})}{1 + \exp(\mathbf{x}'_i \mathbf{b})} \right)^{m_i - y_i} d\mathbf{b}}. \quad (37.12)$$

Probit Regression

The probit regression model is very similar to the logistic regression where we use a standard normal distribution function $\Phi(\cdot)$ as the link function. The probit link $h(p) = \Phi^{-1}(p) = \mathbf{x}'\beta$ is commonly used in practice. If each response y_i independently follows a Bernoulli distribution with parameter p_i , then the data distribution for probit model becomes

$$l(\beta | \mathbf{y}) = \prod_{i=1}^n (\Phi(\mathbf{x}'_i \beta))^{y_i} (1 - \Phi(\mathbf{x}'_i \beta))^{1 - y_i}. \quad (37.13)$$

Then we employ prior distribution for β to complete the Bayesian model and obtain the posterior density as

$$\pi(\beta | \mathbf{y}) = \frac{\pi(\beta) \prod_{i=1}^n (\Phi(\mathbf{x}'_i \beta))^{y_i} (1 - \Phi(\mathbf{x}'_i \beta))^{1 - y_i}}{\int_{\mathbf{b}} \pi(\mathbf{b}) \prod_{i=1}^n (\Phi(\mathbf{x}'_i \mathbf{b}))^{y_i} (1 - \Phi(\mathbf{x}'_i \mathbf{b}))^{1 - y_i} d\mathbf{b}}. \quad (37.14)$$

Poisson Regression

Count data is a frequent data type in statistics. In count data, the observations can only take non-negative integer values $\{0, 1, 2, 3, \dots\}$. So naturally, the Poisson model serves as a standard model for count data. Here we assume that response variable y_i follows a Poisson distribution with parameter $\mu_i (> 0)$, $i = 1, 2, \dots, n$. Note that $\mathbb{E}(y_i | \mu_i) = \mu_i$ for a Poisson model. Then we choose a logarithmic link function to relate the mean to the linear predictor: $h(\mu) = \log(\mu) = \mathbf{x}'\beta$. The model likelihood is then expressed as

$$l(\boldsymbol{\beta} | \mathbf{y}) = \prod_{i=1}^n \frac{1}{y_i!} \exp(-\exp(\mathbf{x}'_i \boldsymbol{\beta})) (\exp(\mathbf{x}'_i \boldsymbol{\beta}))^{y_i}. \quad (37.15)$$

We specify a prior for $\boldsymbol{\beta}$ to complete the Bayesian formulation of modelling and obtain the posterior density as

$$\pi(\boldsymbol{\beta} | \mathbf{y}) = \frac{\pi(\boldsymbol{\beta}) \prod_{i=1}^n \frac{1}{y_i!} \exp(-\exp(\mathbf{x}'_i \boldsymbol{\beta})) (\exp(\mathbf{x}'_i \boldsymbol{\beta}))^{y_i}}{\int_{\mathbf{b}} \pi(\mathbf{b}) \prod_{i=1}^n \frac{1}{y_i!} \exp(-\exp(\mathbf{x}'_i \mathbf{b})) (\exp(\mathbf{x}'_i \mathbf{b}))^{y_i} d\mathbf{b}}. \quad (37.16)$$

37.9 Bayesian Computation

Computations of various posterior quantities are needed in a Bayesian analysis. For example, posterior mean is popularly used as a standard Bayes estimate of a parameter. Posterior mode and posterior median are also used in some cases.

Posterior variance is computed to find the accuracy of the estimate, and posterior quantiles are used to find the Bayesian credible regions. As we know, posterior distribution can be obtained in a standard form by the use of conjugate priors. Posterior quantities such as above are computed very easily in these cases. However, we do not always use conjugate priors in practice and sometimes they do not even exist. If conjugate priors are not used, we often encounter a situation where posterior distributions are not in a standard form. We present a simple example of univariate case in below.

Example

Let us consider the successful passing rates $\mathbf{y} = (y_1, y_2, \dots, y_n)'$ of a football team in its past n matches. Assume that each y_i independently follows a beta distribution with parameters $(\alpha, 2)$, $y_i \in (0, 1)$, $\alpha > 0$, $i = 1, 2, \dots, n$. Unfortunately, there are not any useful conjugate priors available for this model. If we assign a gamma (b_1, b_2) prior for α (a standard choice), the posterior density function becomes: ◀

$$\begin{aligned} \pi(\alpha | \mathbf{y}) &= \frac{l(\alpha | \mathbf{y}) \pi(\alpha)}{m(\mathbf{y})} \\ &= \frac{\left\{ \prod_{i=1}^n \text{Beta}(\alpha, 2)^{-1} y_i^{\alpha-1} (1 - y_i) \right\} \times \Gamma(b_1)^{-1} b_2^{b_1} \alpha^{b_1-1} e^{-b_2 \alpha}}{\int_{\alpha_0} \left\{ \prod_{i=1}^n \text{Beta}(\alpha_0, 2)^{-1} y_i^{\alpha_0-1} (1 - y_i) \right\} \times \Gamma(b_1)^{-1} b_2^{b_1} \alpha_0^{b_1-1} e^{-b_2 \alpha_0} d\alpha_0} \\ &= \text{Const} \times \left\{ \alpha^n (\alpha + 1)^n \prod_{i=1}^n y_i^\alpha \right\} \times \alpha^{b_1-1} e^{-b_2 \alpha}, \end{aligned} \quad (37.17)$$

which does not belong to any standard family of probability distributions.

In such cases, we need different techniques to compute posterior quantities. In many cases these quantities of interest involved are obtained by integration. Even though numerical integration methods (e.g. integrate, R2Cuba packages in R software) can be used for approximation, they fail to provide accurate estimates when the number dimension of the integral gets high. In this section, we shall briefly describe different computing techniques that are needed for a Bayesian analysis.

37.9.1 Laplace Approximation

Suppose we need to compute the posterior mean of $h(\boldsymbol{\theta})$, which is expressed as

$$\mathbb{E}(h(\boldsymbol{\theta}) | \mathbf{y}) = \int_{\boldsymbol{\theta}} h(\boldsymbol{\theta}) \pi(\boldsymbol{\theta} | \mathbf{y}) d\boldsymbol{\theta} = \frac{\int_{\boldsymbol{\theta}} h(\boldsymbol{\theta}) l(\boldsymbol{\theta} | \mathbf{y}) \pi(\boldsymbol{\theta}) d\boldsymbol{\theta}}{\int_{\boldsymbol{\theta}} l(\boldsymbol{\theta} | \mathbf{y}) \pi(\boldsymbol{\theta}) d\boldsymbol{\theta}}, \quad (37.18)$$

where $l(\boldsymbol{\theta} | \mathbf{y})$ denotes the model likelihood of $\boldsymbol{\theta}$ (a p -dimensional parameter vector) given data \mathbf{y} , $\pi(\boldsymbol{\theta})$ denotes prior density and each of $h(\cdot), f(\cdot), \pi(\cdot)$ is a smooth function. Consider an integral of the form

$$I = \int_{\boldsymbol{\theta}} g(\boldsymbol{\theta}) e^{n\gamma(\boldsymbol{\theta})} d\boldsymbol{\theta}, \quad (37.19)$$

where $\gamma(\cdot)$ is a smooth function with its unique maximum at $\hat{\boldsymbol{\theta}}$. We obtain the Laplace approximation of the integral after expanding $g(\cdot)$ and $\gamma(\cdot)$ in Taylor series about $\hat{\boldsymbol{\theta}}$:

$$I = \left\{ g(\hat{\boldsymbol{\theta}}) + (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})' g'(\hat{\boldsymbol{\theta}}) + \frac{1}{2} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})' \Delta_g(\hat{\boldsymbol{\theta}}) (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) + \dots \right\}$$

$$\begin{aligned} & \times e^{n\gamma(\hat{\theta})} \exp\left(\frac{n}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})' \Delta_{\gamma}(\hat{\boldsymbol{\theta}})(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) + \dots\right) d\boldsymbol{\theta} \\ & = e^{n\gamma(\hat{\theta})} (2\pi)^{p/2} n^{-p/2} |\Delta_{\gamma}(\hat{\boldsymbol{\theta}})|^{-1/2} \left\{ g(\hat{\boldsymbol{\theta}}) + O(n^{-1}) \right\}. \end{aligned} \quad (37.20)$$

where $g'(\cdot)$ and $\gamma'(\cdot)$ denote the first derivative of $g(\cdot)$ and $\gamma(\cdot)$, respectively, and Δ_g and Δ_{γ} denote the Hessian matrices for $g(\cdot)$ and $\gamma(\cdot)$, respectively:

$$\Delta_g(\boldsymbol{\theta}) = \left(\left(\frac{\partial^2}{\partial \theta_i \partial \theta_j} g(\boldsymbol{\theta}) \right) \right), \quad \Delta_{\gamma}(\boldsymbol{\theta}) = \left(\left(\frac{\partial^2}{\partial \theta_i \partial \theta_j} \gamma(\boldsymbol{\theta}) \right) \right). \quad (37.21)$$

Now by letting $q(\cdot)$ as $h(\cdot)$ and 1 for the numerator and denominator of (37.18), respectively, we obtain a first order approximation of $\mathbb{E}(h(\boldsymbol{\theta}) | \mathbf{y})$ as:

$$\mathbb{E}(h(\boldsymbol{\theta}) | \mathbf{y}) \approx h(\hat{\boldsymbol{\theta}}) \left\{ 1 + O(n^{-1}) \right\}. \quad (37.22)$$

Now suppose that $h(\cdot)$ in (37.18) is positive and we let $n\gamma(\boldsymbol{\theta}) = \log h(\boldsymbol{\theta}) + \log l(\boldsymbol{\theta} | \mathbf{y}) + \log \pi(\boldsymbol{\theta})$ for the numerator and $n\gamma^*(\boldsymbol{\theta}) = \log l(\boldsymbol{\theta} | \mathbf{y}) + \log \pi(\boldsymbol{\theta})$ for the denominator in (37.18). Tierney and Kadane [6] proposed the following approximation under these conditions:

$$\begin{aligned} \mathbb{E}(h(\boldsymbol{\theta}) | \mathbf{y}) & \approx \left(\frac{|\Delta_{\gamma}(\hat{\boldsymbol{\theta}})|}{|\Delta_{\gamma^*}(\hat{\boldsymbol{\theta}})|} \right)^{1/2} \exp\left(n(\gamma^*(\hat{\boldsymbol{\theta}}) - \gamma(\hat{\boldsymbol{\theta}}))\right) \\ & \left\{ 1 + O(n^{-2}) \right\}, \end{aligned} \quad (37.23)$$

where $\gamma^*(\cdot)$ attains its maximum at $\tilde{\boldsymbol{\theta}}$ and $\gamma(\cdot)$ attains its maximum at $\hat{\boldsymbol{\theta}}$. There are many other analytic approximations in the literature which we did not include here. This is because the superior alternative computation techniques, such as Metropolis-Hastings algorithm, Gibbs algorithm, etc., have made the above analytic approximation methods less popular.

37.9.2 The E-M Algorithm

Now suppose that $\pi(\boldsymbol{\theta} | \mathbf{y})$ is of a non-standard form and computationally difficult to handle. In E-M algorithm, we augment the observed data \mathbf{y} with unobserved latent data \mathbf{z} such that the augmented posterior density $\pi(\boldsymbol{\theta} | \mathbf{y}, \mathbf{z})$ is easier to handle. E-M algorithm [see 7] is essentially an iterative method to find maximum likelihood estimates of parameters in a statistical model. Here we shall use the algorithm to estimate the maximum a posteriori (MAP). Note that $\pi(\boldsymbol{\theta} | \mathbf{y}) = \pi(\boldsymbol{\theta}, \mathbf{z} | \mathbf{y}) / f(\mathbf{z} | \mathbf{y}, \boldsymbol{\theta})$ where $f(\mathbf{z} | \mathbf{y}, \boldsymbol{\theta})$ denotes the predictive density of \mathbf{z} given \mathbf{y} and $\boldsymbol{\theta}$. From this we have

the following identity:

$$\begin{aligned} \log \pi(\boldsymbol{\theta} | \mathbf{y}) & = \int_{\mathbf{z}} \log \pi(\boldsymbol{\theta}, \mathbf{z} | \mathbf{y}) f(\mathbf{z} | \mathbf{y}, \hat{\boldsymbol{\theta}}^{(i)}) d\mathbf{z} \\ & \quad - \int_{\mathbf{z}} \log f(\mathbf{z} | \mathbf{y}, \boldsymbol{\theta}) f(\mathbf{z} | \mathbf{y}, \hat{\boldsymbol{\theta}}^{(i)}) d\mathbf{z} \\ & = Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}) - H(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}), \end{aligned} \quad (37.24)$$

where $\hat{\boldsymbol{\theta}}^{(i)}$ is an estimate of $\boldsymbol{\theta}$ from the i^{th} step of the iteration. The E-M algorithm essentially involves two steps at each iteration:

E-step: Calculate $Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}) = \mathbb{E}_{\mathbf{z} | \mathbf{y}, \hat{\boldsymbol{\theta}}^{(i)}}(\log \pi(\boldsymbol{\theta}, \mathbf{z} | \mathbf{y}))$

M-step: Maximise $Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)})$ with respect to $\boldsymbol{\theta}$, and obtain $\hat{\boldsymbol{\theta}}^{(i+1)}$ such that

$$Q(\hat{\boldsymbol{\theta}}^{(i+1)}, \hat{\boldsymbol{\theta}}^{(i)}) = \max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(i)}).$$

We start by finding $\mathbf{z}^{(i)} = \mathbb{E}(\mathbf{z} | \mathbf{y}, \hat{\boldsymbol{\theta}}^{(i)})$, (if exact mean is not available, we get an estimate of $\mathbb{E}(\mathbf{z} | \mathbf{y}, \hat{\boldsymbol{\theta}}^{(i)})$ as $\mathbf{z}^{(i)}$). Then we use this $\mathbf{z}^{(i)}$ to maximise $\pi(\boldsymbol{\theta} | \mathbf{y}, \mathbf{z}^{(i)})$ to obtain $\hat{\boldsymbol{\theta}}^{(i+1)}$. Then we find $\mathbf{z}^{(i+1)}$ using $\hat{\boldsymbol{\theta}}^{(i+1)}$ and continue the iterations in this manner. Also note that $\pi(\hat{\boldsymbol{\theta}}^{(i+1)} | \mathbf{y}) \geq \pi(\hat{\boldsymbol{\theta}}^{(i)} | \mathbf{y})$ for each iteration i . Therefore, the E-M algorithm can be expected to converge to a local maximum from any starting point.

37.9.3 Monte Carlo Sampling

Simulation of random processes saw its popularity only after the beginning of computer era in the 1990s. Before that, simulations were only used in a handful of studies [8, 9], mainly in the fields of physics and chemistry. Monte Carlo sampling is named after a famous casino at 'Monte Carlo' in the 1950s. Monte Carlo method is also known as 'Ordinary Monte Carlo' (OMC). Let us consider an expectation $\mu = \mathbb{E}[g(X)]$ of a random variable X , where $g(\cdot)$ is a real valued function on the state space such that this expectation is not available in closed form. If i.i.d. sample X_1, X_2, \dots can be simulated from the same distribution as X , then

$$\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n g(X_i) \quad (37.25)$$

converges almost surely to $\mathbb{E}[g(X)]$. Therefore, for large n , we can use $\hat{\mu}_n$ as an estimate of μ . If $\sigma^2 = \text{Var}[g(X)]$ is finite, then σ^2 can be estimated by $\hat{\sigma}_n^2 = \frac{1}{n} \sum_{i=1}^n (g(X_i) - \hat{\mu}_n)^2$. Now we have the asymptotic distribution of $\hat{\mu}_n$ using the central limit theorem (CLT)

$$\frac{\hat{\mu}_n - \mu}{\hat{\sigma}_n/\sqrt{n}} \xrightarrow[n \rightarrow \infty]{d} \mathcal{N}(0, 1). \tag{37.26}$$

Hence $\hat{\mu}_n \pm z_{\alpha/2} \hat{\sigma}_n/\sqrt{n}$ is an asymptotic $100(1 - \alpha)\%$ confidence interval for μ with $z_{\alpha/2}$ denoting the $100(1 - \alpha/2)\%$ quantile of standard normal distribution. Note that each additional significant figure in the estimate (i.e. a tenfold increase in accuracy) requires a hundredfold increase in the sample size. This is because accuracy is inversely proportional to the square root of the sample size (just like classical statistics). Since computer simulations do not generate random samples, they are rather pseudorandom; we call $\hat{\mu}_n$ as *Monte Carlo approximation* of μ , $\hat{\sigma}_n/\sqrt{n}$ as *Monte Carlo standard error* (MCSE) and n as *Monte Carlo sample size*.

37.9.4 Importance Sampling

Suppose we want to obtain an estimate of $\mu = \mathbb{E}[g(X)]$ and it is difficult or costly to sample from the distribution of X . In that case Monte Carlo sampling will not work and we need alternative strategy. Importance sampling uses the idea that a sample can be generated from any distribution. That means, we can use a sample generated from another distribution (other than distribution of X) to compute an estimate of μ . Assume that the probability distribution of X has a density f and there exists another density f_0 which behaves very similar to f but is easy to sample from. Then

$$\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n g(X_i) \frac{f(X_i)}{f_0(X_i)} \tag{37.27}$$

can be used as an estimate of μ for large n , where (X_1, X_2, \dots, X_n) is an i.i.d. sample generated from f_0 . Also note that

$$\begin{aligned} \mathbb{E}_0[\hat{\mu}_n] &= \frac{1}{n} \sum_{i=1}^n \mathbb{E}_0 \left[g(X_i) \frac{f(X_i)}{f_0(X_i)} \right] = \int g(X) \frac{f(X)}{f_0(X)} f_0(X) dX \\ &= \mathbb{E}[g(X)] = \mu. \end{aligned} \tag{37.28}$$

and consequently $\hat{\mu}_n$ converges to μ almost surely. The density f_0 is called an *importance function* in this context. The rationale behind this will be discussed later. Monte Carlo standard error can be calculated using s_n/\sqrt{n} in the same manner as described in (37.9.3), but by replacing $g(X_i)$ by $g(X_i)f(X_i)/f_0(X_i)$. Note that method of importance sampling also works with a Markov chain X_1, X_2, \dots drawn from f_0 rather than an i.i.d. sample.

So far we have assumed that both f and f_0 are normalised densities. In the situation where we have only unnormalised densities h and h_0 , we will require a different estimator of μ . We define

$$\bar{\mu}_n = \frac{1}{n} \sum_{i=1}^n g(X_i) w(X_i), \text{ where } w(X_i) = \frac{h(X_i)/h_0(X_i)}{\sum_{i=1}^n h(X_i)/h_0(X_i)}. \tag{37.29}$$

This new statistic $\bar{\mu}_n$ can be regarded as an estimate of μ , for large n . We also have

$$\begin{aligned} \bar{\mu}_n &\xrightarrow[n \rightarrow \infty]{a.s.} \frac{\mathbb{E}_0 \left[g(X) \frac{h(X)}{h_0(X)} \right]}{\mathbb{E}_0 \left[\frac{h(X)}{h_0(X)} \right]} = \frac{\int g(X) \frac{h(X)}{h_0(X)} f_0(X) dX}{\int \frac{h(X)}{h_0(X)} f_0(X) dX} \\ &= \frac{C}{C_0} \frac{\int g(X) f(X) dX}{\int f(X) dX} = \mathbb{E}[g(X)], \end{aligned} \tag{37.30}$$

where $f(X) = h(X)/C$ and $f_0(X) = h_0(X)/C_0$ with $C = \int h(X) dX$, $C_0 = \int h_0(X) dX$.

37.9.5 Rejection Sampling

Rejection sampling is a very standard method of stochastic simulation [10, 11]. Consider the problem where we want to simulate from an unnormalised density $g(\cdot)$. Suppose that there exists a density $h(\cdot)$ such that (a) it is easy to simulate a sample from $h(\cdot)$, (b) $h(\cdot)$ is very close to $g(\cdot)$ and (c) for some $M > 0$, we have $g(x) < Mh(x)$ for all x . Then we follow the following steps:

- Step 1.** Generate x from $h(\cdot)$ and generate u from *Uniform*(0, 1).
- Step 2.** If $u \leq \frac{g(x)}{Mh(x)}$, we accept the proposed value x .
- Step 3.** We repeat Steps 1 and 2 until we obtain a sample of size n (here n is a prefixed positive integer).

The components of the sample can be regarded as a sample from the target density $g(\cdot)$. Ideally, the positive constant M should be chosen as small as possible to complete the above sampling algorithm with fewer iterations. Note that if we choose $h(x) = g(x)/c$ where c is the normalising constant of $g(\cdot)$, the optimised value of M would be c and we would have gotten the acceptance probability of the sampling algorithm as 1.

37.9.6 Markov Chain Monte Carlo

Markov chain Monte Carlo (MCMC) came into the picture just after the invention of Monte Carlo sampling in the 1950s. The theory of MCMC shares the same motive as Monte Carlo sampling, that is, to compute expectations, but it uses Markov chains. This overcomes the serious drawback that Monte Carlo sampling has, that is, to draw samples directly from the respective distribution. In many of the Bayesian modelling in practice, posterior distribution is incompletely

or implicitly specified that is difficult to handle (e.g. joint posterior distribution of a parameter vector gets specified by their conditional distributions and marginal distributions). Therefore Monte Carlo sampling cannot be used in these cases.

MCMC draws sample by carefully constructing a Markov chain of relatively longer size. The key idea behind the theory of MCMC is that Markov chain with the same equilibrium distribution can also be used to infer about the quantity of interest. There are many available techniques to construct such a Markov chain, most famous ones are *Metropolis algorithm* [8], *Metropolis-Hastings algorithm* [a generalised version of Metropolis algorithm, 9] and *Gibbs sampler* [a special case of Metropolis-Hastings algorithm, 12]. It is to be noted that Markov chain Monte Carlo techniques can be used in both the paradigms: Bayesian and frequentist. Many of the posterior quantities in a Bayesian analysis require integrating over the posterior distribution of the parameters given the data. In likelihood-based inference, one may need to integrate the likelihood over the conditional distribution of observables given the parameters, or one simply may need to calculate a likelihood with complicated dependence [13–16].

Markov Chains and Stationarity

A sequence of random variables $\{X_n : n \geq 1\}$ is called a *stochastic process*. When these random variables also follow the *Markov property*, i.e. the probability of moving to the next state depends only on the present state and not on the past states, then the sequence $\{X_n : n \geq 1\}$ is called a *Markov chain* (here we only use a discrete-time Markov chain to explain the main ideas). We shall denote the chain $\{X_n : n \geq 1\}$ by $\{X_n\}$ for notational convenience. The set of all possible value of X_i is called the *state space* of the Markov chain, denoted by S . The marginal distribution of X_1 , denoted by $p(x)$, is called the *initial distribution*. For general state space S , the transition probabilities are specified by defining a kernel, viz. transition kernel.

Definition 1 (Transition kernel) Let S be a set and \mathcal{S} a σ -field on S . A transition kernel H is a function from $S \times S$ into $[0, 1]$ such that

- (i) $H(x, A) = \Pr(X_{n+1} \in A | X_n = x)$, $x \in S, A \in \mathcal{S}$.
- (ii) For all $x \in S$, $A \mapsto H(x, A)$ is a probability measure on (S, \mathcal{S}) .
- (iii) For all $A \in \mathcal{S}$, $x \mapsto H(x, A)$ is a measurable function on S .

A transition probability of a Markov chain is called *stationary* if the conditional probabilities $H(x, A) = \Pr(X_{n+1} \in A | X_n = x)$ do not depend on n for any $x \in S, A \in \mathcal{S}$. A Markov chain is called *stationary* (or *invariant* or *time homogeneous*) if the conditional distribution of $(X_{n+1}, X_{n+2}, \dots, X_{n+k})$ given

X_n does not depend on the value of n . It consequently follows that a Markov chain is stationary if and only if the marginal distribution of X_n does not depend on n . An initial distribution is called *stationary* (or *invariant* or *equilibrium*) for some transition probability distribution if the corresponding Markov chain specified by this initial distribution and transition probability distribution is stationary. It is to be noted that transition probability distribution of a Markov chain is stationary if the corresponding Markov chain is stationary, but the converse is not generally true.

Definition 2 Consider a Markov chain $\{X_n\}$ with general state space S and transition kernel H .

- (a) A Markov chain $\{X_n\}$ is ϕ -*irreducible* if there exists a probability distribution ϕ and a positive integer $n \geq 1$ such that the n -step transition density

$$H^{(n)}(x, A) = P(X_{n+1} \in A | X_1 = x) > 0, \quad (37.31)$$

for any $x \in S$ and for all $A \in \mathcal{S}$ with $\phi(A) > 0$. A Markov chain $\{X_n\}$ is *irreducible* if it is ϕ -irreducible for some probability distribution ϕ .

- (b) A Markov chain $\{X_n\}$ is *reversible* with respect to a probability distribution p if it satisfies the following condition:

$$p(dx) H(x, dy) = p(dy) H(y, dx) \text{ for all } x, y \in S. \quad (37.32)$$

- (c) A ϕ -irreducible Markov chain $\{X_n\}$ is *recurrent* if for all $A \in \mathcal{S}$ with $\phi(A) > 0$ we have

$$\mathbb{E}_\phi \left[\sum_{n=1}^{\infty} I(X_{n+1} \in A) \mid X_1 = x \right] = \infty, \text{ for any } x \in A. \quad (37.33)$$

- (d) An irreducible and aperiodic Markov chain $\{X_n\}$ is called *positive recurrent* if it has a stationary probability distribution, i.e. there exists a unique p such that

$$p(A) = \int_S p(x) H(x, A) dx \text{ for all } A \in \mathcal{S}. \quad (37.34)$$

- (e) A Markov chain $\{X_n\}$ with stationary distribution p is *aperiodic* if there does not exist any $d \geq 2$ and disjoint subsets $S_1, S_2, \dots, S_d \subseteq S$ with $H(x, S_{i+1}) = 1$ for all $x \in S_i, i = 1, 2, \dots, d - 1$, and $H(x, S_1) = 1$ for all $x \in S_d$, such that $p(S_1) > 0$.

Theorem 1 Consider a ϕ -irreducible and aperiodic Markov chain $\{X_n\}$ with a general state space S and transition kernel H . If the Markov chain has a stationary distribution p , then the following hold:

- (a) The chain $\{X_n\}$ is p -irreducible and the stationary distribution p is unique (that means, the chain is positive recurrent).
- (b) $\|H^{(n)}(x, \cdot) - p(\cdot)\| = \sup_{A \in \mathcal{S}} |H^{(n)}(x, A) - p(A)| \xrightarrow{n \rightarrow \infty} 0$, for p -a.e. $x \in \mathcal{S}$.
- (c) $H^{(n)}(x, A) \xrightarrow{n \rightarrow \infty} p(A)$, for p -a.e. $x \in \mathcal{S}$ and for any $A \in \mathcal{S}$ with $p(A) > 0$.
- (d) For any real valued function $g(\cdot)$ such that $\mathbb{E}_p[|g(X)|] < \infty$,

$$\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n g(X_i) \xrightarrow[n \rightarrow \infty]{a.s.} \mathbb{E}_p[g(X)] = \int_{\mathcal{S}} g(x) p(x) dx. \tag{37.35}$$

The result in Theorem 1(d) is called the *strong law of large numbers* for Markov chain. This result guarantees that each iterate of an ergodic Markov chain except a first few is approximately distributed as p . However, it does not give any information about how many iterations will be needed to reach the convergence.

Definition 3 (Geometric ergodicity) A Markov chain $\{X_n\}$ with stationary distribution p is *geometrically ergodic* if there exists a constant $\lambda \in [0, 1)$ such that

$$\|H^{(n)}(x, \cdot) - p(\cdot)\| \leq \gamma(x)\lambda^n, \quad \text{for } p\text{-a.e. } x \in \mathcal{S} \text{ and } n \geq 1. \tag{37.36}$$

with $\gamma(x) < \infty$ for p -a.e. $x \in \mathcal{S}$.

We define $\lambda^* = \inf\{\lambda : \exists \gamma(\cdot) \text{ satisfying (37.36)}\}$, i.e. the smallest λ for which there exists a function $\gamma(\cdot)$ that satisfies (37.36). This λ^* is called the *rate of convergence*. The discussion of Markov chains and stationarity for a countable state space is covered in [17].

Theorem 2 (Central Limit Theorem) Consider a ϕ -irreducible and aperiodic Markov chain $\{X_n\}$ on a general state space \mathcal{S} with transition kernel H and a stationary distribution p . Assume that p is also the initial distribution, i.e., $X_1 \sim p$. Then for a real valued function $g(\cdot) : \mathcal{S} \rightarrow \mathbb{R}$ with finite mean $\mu = \mathbb{E}_p[g(X_i)]$ and $\text{Var}_p[g(X_i)] < \infty$, the following hold:

$$\sqrt{n}(\hat{\mu}_n - \mu) \xrightarrow[n \rightarrow \infty]{d} \mathcal{N}(0, \sigma^2), \tag{37.37}$$

where $\sigma^2 = \text{Var}_p[g(X_i)] + \sum_{j=1}^{\infty} \text{Cov}_p[g(X_i), g(X_{i+j})] < \infty$.

If this σ^2 is very large compared to $\text{Var}_p[g(X_i)]$, the MCMC algorithm behaves very inefficiently [18]. We could estimate Monte Carlo variance of $\hat{\mu}_n$ by $\hat{\sigma}_n^2/n$ where

$\hat{\sigma}_n^2 = \frac{1}{n} \sum_{i=1}^n (g(X_i) - \hat{\mu}_n)^2$, but this is expected to be an underestimate of $\text{Var}(\hat{\mu}_n)$ due to the positive autocorrelation in MCMC samples. We define *effective sample size* (ESS) [19] as $ESS = n/\kappa(\mu)$ where $\kappa(\mu)$ is the autocorrelation time for μ :

$$\kappa(\mu) = 1 + 2 \sum_{j=1}^{\infty} \rho_j(\mu), \tag{37.38}$$

with $\rho_j(\mu)$ denoting the autocorrelation at lag j for μ . In practice, $\kappa(\mu)$ is computed by using the corresponding MCMC chain of μ and by truncating the summation when the sample autocorrelations get very small (e.g. below 0.1) [20]. Then we estimate $\text{Var}(\hat{\mu}_n)$ by $\hat{\sigma}_n^2/ESS(\mu)$. Another method of variance estimation is via batching. Let us define

$$\hat{\mu}_{b,n} = \frac{1}{K} \sum_{i=(b-1)K+1}^{bK} g(X_i), \quad b = 1, 2, \dots, B, \tag{37.39}$$

where iterates of the Markov chain (X_i 's) are of length $n = BK$ for a sufficiently large K . Then $\sqrt{K}(\hat{\mu}_{b,n} - \mu)$ converges in distribution to $\mathcal{N}(0, \sigma^2)$, for each $b = 1, 2, \dots, B$. Therefore σ^2 can be estimated by

$$\hat{\sigma}_{\text{bm}}^2 = \frac{K}{B-1} \sum_{b=1}^B (\hat{\mu}_{b,n} - \hat{\mu})^2. \tag{37.40}$$

Here, $\hat{\mu}_{b,n}$'s are called *batch means* and $\hat{\sigma}_{\text{bm}}^2$ is called the *batch means estimator* of σ^2 . It is known that $\hat{\sigma}_{\text{bm}}^2$ is not a consistent estimator of σ^2 [21, 22]. However, $\hat{\sigma}_{\text{bm}}^2$ is strongly consistent for σ^2 if the batch size (K) and number of batches (B) also increase with n (e.g. by setting $B = K = \lfloor n^{1/2} \rfloor$) [23].

Overlapping batch means estimator generally has better finite sample properties than $\hat{\sigma}_{\text{bm}}^2$. Here we define the batch means by $\hat{\mu}_{j,K} = K^{-1} \sum_{i=1}^K g(X_{j+i})$, for $j = 0, 1, \dots, n-K$. The overlapping batch means estimator of σ^2 is defined as

$$\hat{\sigma}_{\text{obm}}^2 = \frac{nK}{(n-K)(n-K+1)} \sum_{j=0}^{n-K} (\hat{\mu}_{j,K} - \hat{\mu})^2. \tag{37.41}$$

This $\hat{\sigma}_{\text{obm}}^2$ is also strongly consistent under some mild conditions. Under geometric ergodicity of the Markov chain and existence of a little more than fourth-order moment, both batch means estimator and overlapping batch means estimator are MSE consistent, i.e. $\mathbb{E}_p(\hat{\sigma}_n^2 - \sigma^2)^2 \rightarrow 0$ as $n \rightarrow \infty$ [24].

A reasonable Monte Carlo standard error estimate of $\hat{\mu}_n$ is given by $\hat{\sigma}_n/\sqrt{n}$, which helps to assess the accuracy of $\hat{\mu}_n$. The estimation of variance for a Markov chain is an important step in a MCMC practice and is discussed in detail in [20, 24] and the references therein. We would also like to emphasise here that we are only being able to cover basic theories and techniques on MCMC. Please refer to the following sources for a detailed treatment on the topic: [17, 18, 25, 26]. We describe some well-known MCMC algorithms in the next few sections.

Metropolis-Hastings Algorithm

In MCMC, we are interested in update techniques to construct a Markov chain $\{X_n\}$ that preserves a specified distribution p in each update. In Bayesian analysis, this specified distribution will always be the *posterior distribution*. Let us consider a general state space S , and let Q be a transition kernel with density q such that

$$q(x, y) \geq 0, \text{ for all } x, y \in S \quad (37.42)$$

$$\text{and } Q(x, S) = \int_S q(x, y) dy = 1, x \in S. \quad (37.43)$$

In practice, we chose Q in such a way that it is easy to sample from. This transition density $q(x, y)$ is sometimes referred to as *proposal density* or *candidate density*. We start the algorithm by initialising the chain at a possible state x_1 such that $p(x_1) > 0$. Then we perform the following steps for $m = 2, \dots, n$.

Step 1. Draw a *proposal value* (or *candidate value*) y from density $q(x_{m-1}, y)$.

Step 2. Compute the *Hastings ratio*

$$\rho(x_{m-1}, y) = \frac{p(y) q(y, x_{m-1})}{p(x_{m-1}) q(x_{m-1}, y)}. \quad (37.44)$$

and acceptance probability $\alpha(x_{m-1}, y) = \min\{\rho(x_{m-1}, y), 1\}$.

Step 3. Set the next iterate X_m as

$$x_m = \begin{cases} y & \text{with probability } \alpha(x_{m-1}, y), \\ x_{m-1} & \text{with probability } 1 - \alpha(x_{m-1}, y). \end{cases} \quad (37.45)$$

Observe that $\{X_n\}$ is a Markov chain with transition kernel H with density h such that $h(x, y) = \alpha(x, y) q(x, y)$ for all $x, y \in S$. Also note that $\{X_n\}$ satisfies the condition: $p(x) h(x, y) = p(y) h(y, x)$ for all $x, y \in S$ [27]. This implies that the Markov chain is $\{X_n\}$ *reversible*. Consequently, we have $\int_S p(x) H(x, A) dx = p(A)$, i.e. p is a stationary distribution for transition kernel H . This algorithm is known as the *Metropolis-Hastings algorithm* [9]. Note that it is enough to have an unnormalised version of the density p to run the algo-

rithm. This is particularly useful in Bayesian analysis where it is often difficult to compute the normalisation constant of the posterior distribution. Now if S is irreducible with respect to Q and $p(x) > 0$ for all $x \in S$, the above constructed Markov chain $\{X_n\}$ with transition kernel H and stationary distribution p is also irreducible, and hence ergodic theorem holds.

The *rejection probability* from a point $x \in S$ is defined as

$$r(x) = 1 - \int_S h(x, y) dy, \quad (37.46)$$

and the *expected acceptance rate* of the algorithm is defined as

$$a = \int_S (1 - r(x)) p(x) dx. \quad (37.47)$$

The empirical acceptance rate (i.e. the proportion of proposed updates during the MCMC iterations that are accepted) is used as an estimate of the expected acceptance rate a . For a discrete state space S , we proceed as in the general case. Here we assume the transition probability matrix $Q = ((q(i, j)))$ on the countable state space S . We define the Hasting ratio as

$$\rho(i, j) = \frac{p(j) q(j, i)}{p(i) q(i, j)}$$

and the acceptance probability as $\alpha(i, j) = \min\{\rho(i, j), 1\}$ for all $i, j \in S$.

Metropolis Algorithm

Metropolis algorithm is a special case of Metropolis-Hastings algorithm where the proposal density q is symmetric with respect to its arguments, i.e. $q(x, y) = q(y, x)$ for all states $x, y \in S$ [8]. Then the Hastings ratio in (37.44) reduces to $\rho(x, y) = p(y)/p(x)$, which is now called the *Metropolis ratio*.

Random Walk Metropolis Algorithm

Random walk Metropolis algorithm [28] is a specific case of the Metropolis-Hastings algorithm, where the candidates X_m^* are constructed as $X_m^* = X_{m-1} + \epsilon_m$, with the ϵ_m chosen to be i.i.d. with a symmetric distribution. We also assume that ϵ_m is stochastically independent of X_m . Then $q(X_{m-1}, X_m^*)$ can be expressed as $g(X_m^* - X_{m-1})$ where g denotes the density of ϵ_m . Some popular choices of the proposal density g are $\mathcal{N}(0, \tau^2)$, *Uniform* $(-\delta, \delta)$. The choice of the variance parameter τ^2 in case of a normal proposal is very important in MCMC, and choosing an optimal value of it is itself a different area of research [29, 30]. It is suggested to choose the scale parameter (e.g. τ^2 in the normal proposal) appropriately as it directly influences the overall acceptance rate of the algorithm. If τ^2 is larger than necessary, the algorithm will have high rejection rate, and as a result the chain will move slowly. If τ^2 is smaller

than necessary, the algorithm will have high acceptance rate, but will result in slow mixing within the state space. In both of these two cases, the algorithm will fail to behave efficiently.

Independence Sampler

Independence sampler is also a special case of Metropolis-Hastings algorithm where the proposal density $q(x, y) = q(y)$ and therefore transition kernel $Q(x, \cdot)$ do not depend on its first argument x . The Hastings ratio becomes

$$\rho(x, y) = \frac{p(y)q(x)}{p(x)q(y)}.$$

This algorithm is also called as the independence Metropolis-Hastings algorithm [28]. It is to be noted that the convergence of the Markov chain will be fast if the independence proposal density is a good approximation of the density of the stationary distribution. Hence the practitioner should be careful in choosing the proposal density for getting a better efficiency of the algorithm.

Gibbs Sampler

Around the time when Gibbs sampler was first developed by [12], the algorithm was mostly used in spatial statistics. It was the seminal article [31] that made the Gibbs sampler popular to the Bayesian community making them aware of this incredible method of drawing sample from the posterior distribution. The algorithm is as follows. Let p be the target distribution for a random vector $\mathbf{X} = (X_1, \dots, X_k)'$, $k \geq 2$. We assume that it is possible to generate samples from each full conditional distribution with densities $\{p(X_i | \mathbf{X}_{-i})\}$, where $\mathbf{X}_{-i} = (X_1, \dots, X_{i-1}, X_{i+1}, \dots, X_k)'$. Gibbs sampling is then used to produce sample from the stationary distribution of the Markov chain under some regularity conditions [32], by drawing sample from these full conditional distributions. The steps of Gibbs sampling algorithm are given below. We start the sampling by fixing a set of initial values $(X_2^{(1)}, \dots, X_k^{(1)})$ for parameters (X_2, \dots, X_k) . Then we repeat the following steps for $m = 2, \dots, n$.

Step 1. Draw $X_1^{(m)}$ from $p(X_1 | X_2^{(m-1)}, X_3^{(m-1)}, \dots, X_k^{(m-1)})$.

Step 2. Draw $X_2^{(m)}$ from $p(X_2 | X_1^{(m)}, X_3^{(m-1)}, \dots, X_k^{(m-1)})$.

⋮
⋮

Step k. Draw $X_k^{(m)}$ from $p(X_k | X_1^{(m)}, X_2^{(m)}, \dots, X_{k-1}^{(m)})$.

This algorithm defines a Markov chain with transition probability from $\mathbf{X}^{(m-1)}$ to $\mathbf{X}^{(m)}$ as

$$\prod_{i=1}^k p(X_i^{(m)} | X_1^{(m)}, \dots, X_{i-1}^{(m)}, X_{i+1}^{(m-1)}, \dots, X_k^{(m-1)}).$$

The following condition, viz. the *positivity condition* [33], is needed to explain the convergence properties of Gibbs sampler without resorting to Metropolis-Hastings algorithm.

Definition 4 (Positivity condition) Suppose that $p(x_1, \dots, x_p)$ is the joint probability density of a random vector $\mathbf{X} = (X_1, \dots, X_p)'$ and $p^{(i)}(x_i)$ denotes the marginal probability density of X_i , $i = 1, \dots, p$. If $p^{(i)}(x_i) > 0$ for every $i = 1, \dots, p$ implies that $p(x_1, \dots, x_p) > 0$, then the joint density p is said to satisfy the *positivity condition*.

Under the positivity condition, it can be shown that the simulated Markov chain $\{\mathbf{X}^{(m)} : m = 1, \dots, n\}$ is irreducible. This in turn validates that the Markov chain has a stationary distribution which is same as the joint posterior distribution p . Thus after discarding all draws from the burn-in period $m = 1, \dots, n_{burn}$ (here n_{burn} is an iteration number specified by the practitioner after which the Markov chain would have approximately distributed as the stationary distribution). Summaries of the distribution can be obtained by using the sample $\{\mathbf{X}^{(m)} : m = n_{burn} + 1, \dots, n\}$, for a sufficiently large n .

It is important to note that the above full conditionals have sufficient information to uniquely determine the joint density. *Brook's lemma* [34] (also known as *the factorisation theorem*) demonstrates that given the full conditional density $p(x_i | \mathbf{x}_{-i})$, we can determine the joint density $p(x_1, x_2, \dots, x_p)$ of a random vector $\mathbf{X} = (X_1, X_2, \dots, X_p)'$.

Lemma 1 (Brook's lemma) *The joint density of a random vector $\mathbf{X} = (X_1, X_2, \dots, X_p)'$ can be determined up to a constant by its full conditionals as*

$$\begin{aligned} p(x_1, \dots, x_p) &= \frac{p(x_1 | x_2, \dots, x_p)}{p(x_{10} | x_2, \dots, x_p)} \cdot \frac{p(x_2 | x_{10}, \dots, x_p)}{p(x_{20} | x_{10}, \dots, x_p)} \\ &\dots \dots \frac{p(x_p | x_{10}, \dots, x_{p-1,0})}{p(x_{p0} | x_{10}, \dots, x_{p-1,0})} \\ &\cdot p(x_{10}, \dots, x_{p0}), \end{aligned} \tag{37.48}$$

where $\mathbf{x} = (x_1, x_2, \dots, x_p)'$ and $\mathbf{x}_0 = (x_{10}, x_{20}, \dots, x_{p0})'$ are two fixed points in the support of $p(x_1, x_2, \dots, x_p)$.

It can also be shown that Gibbs sampling algorithm is a special case of Metropolis-Hastings algorithm. Let us consider the block updating of \mathbf{X} to \mathbf{X}^* where only one component X_i is getting updated for some fixed i , i.e. $\mathbf{X}^* = (X_1, \dots, X_{i-1}, X_i^*, X_{i+1}, \dots, X_p)$. We factor the joint density of \mathbf{X} as

$$p(\mathbf{x}) = p_1(\mathbf{x}_{-i}) p_2(x_i | \mathbf{x}_{-i}),$$

where \mathbf{x}_{-i} denotes the parameter vector without the component X_i , p_1 denotes the marginal posterior density of \mathbf{x}_{-i} and p_2 denotes full conditional density of X_i . Now in Gibbs sampling, we generate x_i^* from $p_2(x_i | \mathbf{x}_{-i})$ which can be considered as the proposal distribution in Metropolis-Hastings algorithm. The corresponding Hastings ratio is

$$\begin{aligned}\rho(\mathbf{x}, \mathbf{x}^*) &= \frac{p(\mathbf{x}^*) p_2(x_i | \mathbf{x}_{-i})}{p(\mathbf{x}) p_2(x_i^* | \mathbf{x}_{-i})} \\ &= \frac{p_1(\mathbf{x}_{-i}) p_2(x_i^* | \mathbf{x}_{-i}) p_2(x_i | \mathbf{x}_{-i})}{p_1(\mathbf{x}_{-i}) p_2(x_i | \mathbf{x}_{-i}) p_2(x_i^* | \mathbf{x}_{-i})} = 1,\end{aligned}$$

implying that the proposal values are always accepted in Gibbs sampling.

Slice Sampler

Slice sampler is another popular MCMC algorithm where we use auxiliary variables [35]. Suppose we want to simulate a sample (X_1, X_2, \dots, X_n) from an unnormalised density $g(\cdot)$. In a Bayesian analysis, $g(\cdot)$ will be the unnormalised posterior, that is, the product of likelihood and the prior density. We consider an auxiliary random variable Y with $Y | X = x$ following $Uniform(0, g(x))$. Then the joint density of Y and X is

$$p(x, y) \propto g(x) \frac{1}{g(x)} I(y < g(x)) = I(y < g(x)), \quad (37.49)$$

where $I(y < g(x))$ takes the value 1 if $y < g(x)$ and 0 otherwise. This joint density is essentially a uniform density with the support $0 < Y < g(X)$. The full conditionals for a Gibbs sampler will be the following:

1. Generate Y from $Y | X = x \sim Uniform(0, g(x))$.
2. Note that $X | Y = y$ follows a uniform distribution over the values of x for which $g(x) \geq y$. Generate X from this conditional distribution.

Repeating these two steps for n iterations will obtain a sample $\{(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)\}$ from the joint density $p(x, y)$, and hence (X_1, X_2, \dots, X_n) will be a sample from the marginal density $g(x)$. The name ‘*Slice sampler*’ came from the observation that the line $Y = y$ (which is horizontal to the x -axis) *slices* the unnormalised density curve $g(x)$ into two parts, and we consider only the values of X for which $g(x) \geq y$.

$$\rho\{(\boldsymbol{\theta}_k, \mathcal{M}_k), (\boldsymbol{\theta}_{k^*}^*, \mathcal{M}_{k^*}^*)\} \quad (37.51)$$

$$= \frac{l(\boldsymbol{\theta}_{k^*}^*, \mathcal{M}_{k^*}^* | \mathbf{y}) \pi(\boldsymbol{\theta}_{k^*}^* | \mathcal{M}_{k^*}^*) \pi(\mathcal{M}_{k^*}^*) q(\mathcal{M}_{k^*}^*, \mathcal{M}_k) h_{k,k^*}(\mathbf{u}^* | \boldsymbol{\theta}_{k^*}^*)}{l(\boldsymbol{\theta}_k, \mathcal{M}_k | \mathbf{y}) \pi(\boldsymbol{\theta}_k | \mathcal{M}_k) \pi(\mathcal{M}_k) q(\mathcal{M}_k, \mathcal{M}_{k^*}^*) h_{k,k^*}(\mathbf{u} | \boldsymbol{\theta}_k)} \left| \frac{\partial g(\boldsymbol{\theta}_k, \mathbf{u})}{\partial(\boldsymbol{\theta}_k, \mathbf{u})} \right|, \quad (37.52)$$

where $q(\mathcal{M}_k, \mathcal{M}_{k^*}^*)$ denotes the transition probability from model \mathcal{M}_k to model $\mathcal{M}_{k^*}^*$. We accept the proposed update $(\boldsymbol{\theta}_{k^*}^*, \mathcal{M}_{k^*}^*)$ with probability $\alpha\{(\boldsymbol{\theta}_k, \mathcal{M}_k), (\boldsymbol{\theta}_{k^*}^*, \mathcal{M}_{k^*}^*)\}$.

Repeating the above steps for n iterations will obtain a sample which can then be used to derive the posterior sum-

Reversible Jump Markov Chain Monte Carlo

The reversible jump Markov chain Monte Carlo (RJMCMC) is an extension of Metropolis-Hastings MCMC sampling algorithm where the dimension of the parameter space is also allowed to be updated in different MCMC iterations [36]. We will describe this algorithm in a Bayesian context. Suppose that we have a countable collection of competing models $\{\mathcal{M}_k, k \in \mathcal{K}\}$ for observed data \mathbf{y} . Let each model \mathcal{M}_k have an n_k dimensional parameter vector, denoted by $\boldsymbol{\theta}_k \in S_k \subset \mathbb{R}^{n_k}$. It is to be noted that n_k 's are allowed to vary for different models. Then the posterior density of $(\boldsymbol{\theta}_k, \mathcal{M}_k)$ is obtained as

$$\begin{aligned}\pi(\boldsymbol{\theta}_k, \mathcal{M}_k | \mathbf{y}) \\ = \frac{l(\boldsymbol{\theta}_k, \mathcal{M}_k | \mathbf{y}) \pi(\boldsymbol{\theta}_k | \mathcal{M}_k) \pi(\mathcal{M}_k)}{\sum_{m \in \mathcal{K}} \int l(\boldsymbol{\theta}'_m, \mathcal{M}_m | \mathbf{y}) \pi(\boldsymbol{\theta}'_m | \mathcal{M}_m) \pi(\mathcal{M}_m) d\boldsymbol{\theta}'_m}\end{aligned} \quad (37.50)$$

where $l(\boldsymbol{\theta}_k, \mathcal{M}_k | \mathbf{y})$ denotes the corresponding model likelihood, $\pi(\boldsymbol{\theta}_k | \mathcal{M}_k)$ denotes the prior density of $\boldsymbol{\theta}_k$ under model \mathcal{M}_k and $\pi(\mathcal{M}_k)$ denotes the prior density of model \mathcal{M}_k . This joint posterior density $\pi(\boldsymbol{\theta}_k, \mathcal{M}_k | \mathbf{y})$ is considered to be the target density in RJMCMC, and the new state space is given by $S = \bigcup_{k \in \mathcal{K}} (\{\mathcal{M}_k\}, S_k)$. RJMCMC is essentially a strategy of constructing a Markov chain on this general state space S with the joint posterior distribution as its stationary distribution. RJMCMC uses the following steps at each iteration.

- Step 1.** Suppose that $(\boldsymbol{\theta}_k, \mathcal{M}_k)$ denotes the current state of the chain and n_k denotes the length of $\boldsymbol{\theta}_k$. We generate a candidate model $\mathcal{M}_{k^*}^*$ with transition probability $q(\mathcal{M}_k, \mathcal{M}_{k^*}^*)$.
- Step 2.** A random vector \mathbf{u} is generated from a known density $h_{k,k^*}(\mathbf{u} | \boldsymbol{\theta}_k)$.
- Step 3.** We map $(\boldsymbol{\theta}_k, \mathbf{u})$ to $(\boldsymbol{\theta}_{k^*}^*, \mathbf{u}^*)$ by defining a bijective function $g(\cdot)$ such that $g(\boldsymbol{\theta}_k, \mathbf{u}) = (\boldsymbol{\theta}_{k^*}^*, \mathbf{u}^*)$ and $n_k + \text{length}(\mathbf{u}) = n_{k^*} + \text{length}(\mathbf{u}^*)$.
- Step 4.** The acceptance probability $\alpha\{(\boldsymbol{\theta}_k, \mathcal{M}_k), (\boldsymbol{\theta}_{k^*}^*, \mathcal{M}_{k^*}^*)\}$ of the candidate value $(\boldsymbol{\theta}_{k^*}^*, \mathcal{M}_{k^*}^*)$ is $\min(1, \rho\{(\boldsymbol{\theta}_k, \mathcal{M}_k), (\boldsymbol{\theta}_{k^*}^*, \mathcal{M}_{k^*}^*)\})$ where

maries. The dimension matching in Step 2 implies that the constructed Markov chain is reversible and hence converges to the stationary distribution as in Metropolis-Hastings algorithm.

37.10 Monitoring Convergence

Although convergence rates are available for most MCMC algorithms owing to the active research in this field, they cannot be directly used in practice to know when a chain has converged since the rates are typically available up to an arbitrary constant [37]. Practitioners usually take the help of the diagnostic tools to justify the convergence of the chain. However, it should be noted that diagnostic tools cannot prove convergence of MCMC algorithms as they use only a finite subset of the chain. It is to be noted that the chains obtained from an MCMC algorithm reach the stationary distribution only after a finite number of iterations. Because of this, we truncate these chains from the left to alleviate the effect of starting values and to keep only the part of the chain which can be taken as a sample from the stationary distribution. We call the truncated part as the *burn-in* period.

Estimating the burn-in period is an important aspect of monitoring convergence. If the burn-in is taken too small, we will have some draws left in the sample which cannot be taken as representatives from the stationary distribution; hence, the standard error of the parameter estimate will unnecessarily increase. If the burn-in period is too large, then we are essentially throwing away data which could have been used to improve accuracy of the parameter estimate.

37.10.1 Geweke Diagnostic

Assume that $\{\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(n)}\}$ is an MCMC sample for a scalar parameter θ and $g(\theta)$ is the posterior quantity of interest. Let n_0 denote the iteration number where we want to check whether the chain has converged. We define $A = \{n_0 + 1, n_0 + 2, \dots, n_0 + n_A\}$, $B = \{n_0 + n^*, n_0 + n^* + 1, \dots, n\}$ where $1 < n_A < n^* < n_1$, $n_B = n_1 - n^* + 1$, $n_A + n_B < n_1$ and $n_0 + n_1 = n$. Let

$$\bar{g}_A = \frac{1}{n_A} \sum_{i \in A} g(\theta^{(i)}), \quad \bar{g}_B = \frac{1}{n_B} \sum_{i \in B} g(\theta^{(i)}), \quad (37.53)$$

and let S_g^A and S_g^B denote consistent spectral density estimates for $\{g(\theta^{(i)}) : i \in A\}$ and $\{g(\theta^{(i)}) : i \in B\}$, respectively. Then

$$Z_n = \frac{\bar{g}_A - \bar{g}_B}{\sqrt{S_g^A/n_A + S_g^B/n_B}} \xrightarrow[n \rightarrow \infty]{d} \mathcal{N}(0, 1), \quad (37.54)$$

provided that the sequence $g(\theta^{(i)})$ is stationary and the ratios n_A/n and n_B/n are fixed [38]. Using the above property, we can test the equality of mean at different windows A and B of the sample and reject the null hypothesis if $|Z_n|$ is larger than the critical point. This would indicate that the chain has

not converged by iteration n_0 . Geweke [38] suggested n_A and n_B to be $n_1/10$ and $n_1/2$. At first, the test is applied to the whole chain by considering $n_0 = 0$. If the test is rejected, we sequentially apply the test for different values of n_0 (e.g. $n/10, n/5, 3n/10, 2n/5$, etc.). If the test gets rejected at all the values of n_0 , this would indicate that the chain has not converged.

37.10.2 The Serial Correlation and the Effective Sample Size (ESS)

There is another factor to be considered while drawing inference using a Markov chain. Each component of an MCMC sample, after attaining convergence, follows the target distribution (which is the posterior distribution in case of a Bayesian analysis). Since the sample draws are correlated within themselves, the efficiency of the estimates using the correlated sample is usually lesser than what we would have obtained by using an independent sample. For instance, if we wish to reduce the standard error of an estimator by a factor of 10, we would require to increase the sample size by a factor of 100. However, sample size should be increased by a larger number if we use a correlated sample instead. In such cases, we need to first compute the effective sample size $ESS = n/\kappa$ where κ is the autocorrelation time for the corresponding parameter and can be computed from (37.38) and then increase the sample size by a factor of $100/ESS$ to reduce the standard error by a factor of 10.

37.10.3 Gelman-Rubin Diagnostic

Convergence of every quantity of interest should be monitored while running an MCMC algorithm (e.g. some parametric function, logarithm of posterior density, etc.). Gelman-Rubin diagnostic [3] assesses convergence by checking both mixing and stationarity of the simulated Markov chain. At first, one should simulate m number of chains for the parameter of interest, each of length n . Then, for each scalar parameter θ , we will have the chains $\{\theta_{ij} : i = 1, 2, \dots, n; j = 1, 2, \dots, m\}$. We compute the between-sequence variance B and within-sequence variance W as the following

$$B = \frac{n}{m-1} \sum_{j=1}^m (\bar{\theta}_{.j} - \bar{\theta}_{..})^2 \text{ and } W = \frac{1}{m} \sum_{j=1}^m s_j^2, \quad (37.55)$$

where $\bar{\theta}_{.j} = \frac{1}{n} \sum_{i=1}^n \theta_{ij}$, $\bar{\theta}_{..} = \frac{1}{n} \sum_{j=1}^m \bar{\theta}_{.j}$ and $s_j^2 = \sum_{i=1}^n (\theta_{ij} - \bar{\theta}_{.j})^2$. We estimate the marginal posterior variance $\text{Var}(\theta | y)$ by

$$\widehat{\text{Var}}(\theta | \mathbf{y}) = \frac{n-1}{n}W + \frac{1}{n}B. \quad (37.56)$$

The estimator $\widehat{\text{Var}}(\theta | \mathbf{y})$ overestimates $\text{Var}(\theta | \mathbf{y})$ if the initial distribution is overdispersed. However, the estimator is unbiased if the chain is stationary or in the limit as $n \rightarrow \infty$. On the other hand, W underestimates $\text{Var}(\theta | \mathbf{y})$ for every finite positive integer values of n , but is asymptotically unbiased as n approaches ∞ . Potential scale reduction factor \hat{R} of the distribution of θ is defined as

$$\hat{R} = \left(\frac{\widehat{\text{Var}}(\theta | \mathbf{y})}{W} \right)^{1/2}, \quad (37.57)$$

which declines to 1 in the limit as $n \rightarrow \infty$. The convergence of the chain is monitored by value of \hat{R} . In practice, a higher value of \hat{R} indicates that the practitioner should continue the simulation to improve the inference on the target distribution of θ . Gelman et al. [3] also suggests to always start by simulating at least two chains and then split each chain in half (after discarding the draws from the burn-in period), so that $m \geq 4$.

37.10.4 Model Robustness and Sensitivity Analysis

Bayesian modelling assumes every aspect of a model including the likelihood and the prior(s). So it is not surprising that the inference drawn from fitting of such models to the data is prone to be dependent on these assumptions. To check robustness, a standard approach is to study the sensitivity in the estimated posterior quantities of a Bayesian analysis when we make small changes in the likelihood or prior or other parts of the model. If the perturbations in parts of the model do not lead to effecting the conclusions, we say that the data set is informative enough for the model assumptions. If the conclusions get affected, then we rethink about the model assumptions and/or collect more data.

Example

We take an example of a beta-binomial model with observed data y following a binomial distribution with parameters n and p ; the parameter p is given a $\text{beta}(a, b)$ prior. To check the sensitivity, we increase or decrease a and b separately (and together), and then we compute the posterior quantities of interest in each of the cases. ◀

Sensitivity analysis requires further sampling, which may be time-consuming and costly in many practical scenarios. However, we can avoid the additional sampling in some situations. Suppose we have a data set $\mathbf{y} = (y_1, y_2, \dots, y_n)$

and we model it using the likelihood $l(\theta | \mathbf{y}) = \prod_{i=1}^n l(\theta | y_i)$. We want to investigate the effect of removing the observation y_j from the data set. Assume that $\{\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(N)}\}$ is a sample from the posterior distribution with density $p(\theta | \mathbf{y})$. If posterior mean of $g(\theta)$ is the posterior quantity of interest, then we estimate it as

$$\hat{\mathbb{E}}(g(\theta) | \mathbf{y}) = \sum_{i=1}^N g(\theta^{(i)})/N.$$

After deleting the j -th observation from the data set \mathbf{y} , the modified posterior density can be expressed as

$$p^*(\theta | \mathbf{y}) \propto l(\theta | \mathbf{y}_{-j})p(\theta) \propto \frac{p(\theta | \mathbf{y})}{l(\theta | y_j)}.$$

We estimate the posterior mean of $g(\theta)$ after deleting y_j as

$$\hat{\mathbb{E}}^*(g(\theta) | \mathbf{y}) = \frac{\sum_{i=1}^N g(\theta^{(i)})/l(\theta | y_j)}{\sum_{i=1}^N 1/l(\theta | y_j)}, \quad (37.58)$$

using the importance sampling estimator. Thus we can obtain the posterior quantities under such cases without additional sampling.

37.10.5 Posterior Predictive Checking

Assessing the fit of the model to the data is an important part of statistical modelling. Therefore, in every Bayesian analysis, it is recommended to check the quality of the fit. If a model fits well to the data, the replicated data simulated from the model is expected to be similar as the observed data. Let \mathbf{y}^{rep} denote a replication as observed data \mathbf{y} that also could have been observed with the same model and $\tilde{\mathbf{y}}$ denote future observations. The posterior predictive density is defined as

$$p(\mathbf{y}^{\text{rep}} | \mathbf{y}) = \int l(\theta | \mathbf{y}^{\text{rep}})p(\theta | \mathbf{y})d\theta. \quad (37.59)$$

The Bayesian p -value, denoted by p_B , is defined as

$$p_B = \Pr(T(\mathbf{y}^{\text{rep}}, \theta) \geq T(\mathbf{y}, \theta) | \mathbf{y}), \quad (37.60)$$

where $T(\mathbf{y}, \theta)$ is a scalar function of both the observed data \mathbf{y} and the parameter θ . In practice, if we have an MCMC sample for θ of length n : $\{\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(n)}\}$, we draw \mathbf{y}^{rep} for each iteration $\theta^{(i)}$, and consequently we have a sample $\{(\mathbf{y}^{\text{rep},(i)}, \theta^{(i)}) : i = 1, 2, \dots, n\}$ from the joint posterior distribution with density $p(\mathbf{y}^{\text{rep}}, \theta | \mathbf{y})$. The Bayesian p -value p_B is then estimated as the proportion of the n iterations such that $T(\mathbf{y}^{\text{rep},(i)}, \theta^{(i)}) \geq T(\mathbf{y}, \theta^{(i)})$, $i = 1, 2, \dots, n$. The function $T(\mathbf{y}, \theta)$ should be chosen to represent relevant aspects of the

model fitting. Choosing of the functional $T(\mathbf{y}, \theta)$ is described in detail by [3]. If the model is nearly true, p_B should be close to 0.5. If p_B is estimated as close to 0 or 1, the model is likely to be poor because of its predictions that do not fit the observed data. Under the model, (i) a scatterplot of the values $T(\mathbf{y}^{\text{rep.}(i)}, \theta^{(i)})$ against $T(\mathbf{y}, \theta^{(i)})$ should be symmetric about the 45° line, and (ii) a histogram of the differences $T(\mathbf{y}, \theta^{(i)}) - T(\mathbf{y}^{\text{rep.}(i)}, \theta^{(i)})$ should include 0. A deviation from the above expected outcomes should indicate that the assumed model possesses poor predictive ability.

37.11 Model Comparison

Model comparison is a key practice in Bayesian modelling. As we already know, all the models usually applied to the data are wrong, but among them some can be less wrong and may be informative for making comparisons between several models. Model comparison methods usually focus on predictive ability of a model while penalising for model complexity. Over the years, this topic has been an active area of research, and numerous model comparison methods have been developed. However, there is no unanimity among the Bayesian practitioners about the most appropriate model comparison method. We will not be able to go in much depth in this topic and will only be providing brief descriptions on the methods. Let us start by assuming that we have L number of competing models $\{\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_L\}$.

37.11.1 Akaike Information Criterion (AIC)

Akaike information criterion [AIC, 39] is undoubtedly the most popular model comparison method. AIC is defined as

$$\text{AIC} = -2 \log l(\hat{\theta}_{\text{mle}} | \mathbf{y}) + 2p, \quad (37.61)$$

where $l(\theta | \mathbf{y})$ denotes the model likelihood for data \mathbf{y} and parameter θ . The dimension of the parameter θ is denoted by p , and the maximum likelihood estimate is denoted by $\hat{\theta}_{\text{mle}}$. The second term in the expression of AIC stands for a bias correction for the deviance (a measure of predictive accuracy of a model)

$$D(\theta) = -2 \log l(\theta | \mathbf{y}) \quad (37.62)$$

computed at the point $\hat{\theta}_{\text{mle}}$. In fact, the approximation in AIC is based on asymptotic normality of posterior distribution. However, when the sample size is small, AIC tends to favour models with higher number of parameters. We calculate AIC for each of the L competing models $\{\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_L\}$ and choose the one with the minimum AIC value. Under some conditions, AIC has been shown to be asymptotically

equal to leave-one-out cross-validation (LOO-CV) when it is computed using the maximum likelihood estimate [40].

37.11.2 Deviance Information Criterion (DIC)

The computation of AIC or BIC requires the number of parameters to apply the bias correction. In Bayesian analyses, the number of parameters is often not clear, and the parameters are constrained by the prior in some manner. *Deviance information criterion* [DIC, 41] is defined as

$$\text{DIC} = -2 \log l(\hat{\theta} | \mathbf{y}) + 2p_d \quad (37.63)$$

where the bias correction term

$$p_D = -2 \mathbb{E}_{\theta | \mathbf{y}}[\log l(\theta | \mathbf{y})] + 2 \log l(\hat{\theta} | \mathbf{y})$$

is computed as

$$\hat{p}_D = -\frac{2}{n} \sum_{i=1}^n \log l(\theta^{(i)} | \mathbf{y}) + 2 \log l(\hat{\theta} | \mathbf{y}), \quad (37.64)$$

where $\{\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(n)}\}$ denotes an MCMC sample for θ of length n and $\hat{\theta}$ denotes the estimate of posterior mean or posterior mode. An alternative version of DIC, viz. $\text{DIC}_{\text{alt}} = -2 \log l(\hat{\theta} | \mathbf{y}) + 2p_{D,\text{alt}}$, is also available [42] that makes a modification in the bias correction term

$$p_{D,\text{alt}} = 2 \text{Var}_{\theta | \mathbf{y}}[\log l(\theta | \mathbf{y})]$$

which is computed as

$$\hat{p}_{D,\text{alt}} = \frac{2}{n-1} \sum_{i=1}^n \left(\log l(\theta^{(i)} | \mathbf{y}) - \frac{1}{n} \sum_{j=1}^n \log l(\theta^{(j)} | \mathbf{y}) \right)^2. \quad (37.65)$$

We compute DIC for each model and prefer the model with smallest DIC value. It is to be noted that DIC is numerically more stable in practice than DIC_{alt} . But the bias correction term corresponding to the alternative version has the advantage of being always positive. Under some conditions, DIC has been shown to be asymptotically equal to LOO-CV using plug-in predictive densities [43].

37.11.3 Widely Applicable Information Criterion (WAIC)

Assuming that data \mathbf{y} can be partitioned into d data points such that $\mathbf{y} = (y_1, y_2, \dots, y_d)'$ and joint likelihood $l(\theta | \mathbf{y}) = \prod_{i=1}^d l(\theta | y_i)$, *widely applicable information criteria*, also

known as Watanabe-Akaike information criterion [WAIC, 44], is defined as

$$\text{WAIC} = -2 \sum_{i=1}^d \log \mathbb{E}_{\theta | y} [l(\theta | y_i)] + 2 p_{\text{WAIC}}. \quad (37.66)$$

Two versions of p_{WAIC} are generally used in practice

$$\begin{aligned} p_{\text{WAIC1}} &= 2 \sum_{i=1}^d \left\{ \log \mathbb{E}_{\theta | y} [l(\theta | y_i)] - \mathbb{E}_{\theta | y} [\log l(\theta | y_i)] \right\}, \\ p_{\text{WAIC2}} &= \sum_{i=1}^d \text{Var}_{\theta | y} [\log l(\theta | y_i)]. \end{aligned} \quad (37.67)$$

We compute p_{WAIC} using MCMC draws $\{\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(n)}\}$ from $p(\theta | y)$ as follows:

$$\begin{aligned} \hat{p}_{\text{WAIC1}} &= 2 \sum_{i=1}^d \left\{ \log \left(\frac{1}{n} \sum_{m=1}^n l(\theta^{(m)} | y_i) \right) \right. \\ &\quad \left. - \frac{1}{n} \sum_{m=1}^n \log l(\theta^{(m)} | y_i) \right\}, \\ \hat{p}_{\text{WAIC2}} &= \sum_{i=1}^d \left\{ \frac{1}{n} \sum_{m=1}^n \left(\log l(\theta^{(m)} | y_i) \right. \right. \\ &\quad \left. \left. - \frac{1}{n} \sum_{m=1}^n \log l(\theta^{(m)} | y_i) \right)^2 \right\}. \end{aligned}$$

A model with smaller WAIC value is preferred. The second bias correction term p_{WAIC2} looks similar to $p_{D,\text{alt}}$ but is more stable because of summing up the variances. WAIC has the desirable property of averaging over the posterior distribution rather than conditioning on a point estimate like DIC. WAIC is considered as a fully Bayesian approach for estimating out-of-sample predictive accuracy of a model. Under some conditions, WAIC has been proven to be asymptotically equal to Bayesian LOO-CV [44].

37.11.4 Out-of-Sample Validation

If we have a large data set at hand, we split the data set into two parts – (i) one part is the within sample data $y = (y_1, y_2, \dots, y_d)'$ which is used to fit or train the model, and (ii) the other part is the out-of-sample data $y_{\text{OOS}} = (y_{1,\text{OOS}}, y_{2,\text{OOS}}, \dots, y_{d_{\text{OOS}},\text{OOS}})'$ that is used to validate the model. Suppose $\hat{y}_{\text{OOS}} = (\hat{y}_{1,\text{OOS}}, \hat{y}_{2,\text{OOS}}, \dots, \hat{y}_{d_{\text{OOS}},\text{OOS}})'$ are obtained as model predictions of the data y_{OOS} under model \mathcal{M} . The predictions \hat{y}_{OOS} can be computed as the estimate of

$$\mathbb{E}(y_{\text{OOS}} | y) = \int y_{\text{OOS}} p(y_{\text{OOS}} | y) dy_{\text{OOS}}.$$

which can be computed as $\sum_{m=1}^n y_{\text{OOS}}^{(m)} / n$. Here $y_{\text{OOS}}^{(m)}$ denotes a draw from likelihood $l(\theta^{(m)} | y_{\text{OOS}})$ at each MCMC iteration $m = 1, 2, \dots, n$. We compute the mean squared prediction error (MSPE) [42]

$$\text{MSPE} = \frac{1}{d_{\text{OOS}}} \sum_{i=1}^{d_{\text{OOS}}} (y_{i,\text{OOS}} - \hat{y}_{i,\text{OOS}})^2 \quad (37.68)$$

independently for each of the L models $\{\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_L\}$. The model which corresponds to minimum MSPE is considered to be the best model in the set.

37.11.5 Bayesian Cross-Validation

Cross-validation is a method of measuring predictive ability of a model and is very useful for smaller sample sizes. Leave-one-out cross-validation is the simplest approach for conducting a cross-validation study [3, 42]. In leave-one-out cross-validation, the *score* is defined as

$$\text{score} = \sum_{i=1}^d \log p(y_i | y_{-i}), \quad (37.69)$$

where each $p(y_i | y_{-i})$ can be computed as

$$\frac{1}{n} \sum_{m=1}^n p(y_i | y_{-i}, \theta^{(i,m)}). \quad (37.70)$$

Here $\{\theta^{(i,1)}, \theta^{(i,2)}, \dots, \theta^{(i,n)}\}$ denotes MCMC sample for parameter θ using only $y_{-i} = (y_1, y_2, \dots, y_{i-1}, y_{i+1}, \dots, y_d)'$ as data, $i = 1, 2, \dots, d$. This score statistic underestimates the predictive ability of the model as it uses only $d - 1$ data points while conditioning. The *bias* is expressed as

$$\text{bias} = \sum_{i=1}^d \log p(y_i | y) - \frac{1}{d} \sum_{i=1}^d \sum_{j=1}^d \log p(y_j | y_{-i}), \quad (37.71)$$

where each $p(y_j | y_{-i})$ can be computed as (notice the difference with (37.70))

$$\frac{1}{n} \sum_{m=1}^n p(y_j | y_{-i}, \theta^{(i,m)}). \quad (37.72)$$

The *bias corrected score* is then expressed as $\text{score}_b = \text{score} + \text{bias}$.

K -fold cross-validation can be thought of as an extension of leave-one-out cross-validation where the whole data set y

is partitioned into K subsets $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_K$, and we define the score as

$$\text{score} = \sum_{k=1}^K \log p(\mathbf{y}_k | \mathbf{y}_{-k}). \tag{37.73}$$

where each $p(\mathbf{y}_k | \mathbf{y}_{-k})$ can be computed similarly as

$$\frac{1}{n} \sum_{m=1}^n p(\mathbf{y}_k | \mathbf{y}_{-k}, \boldsymbol{\theta}^{(k,m)}). \tag{37.74}$$

with $\{\boldsymbol{\theta}^{(k,1)}, \boldsymbol{\theta}^{(k,2)}, \dots, \boldsymbol{\theta}^{(k,n)}\}$ denoting an MCMC sample for parameter $\boldsymbol{\theta}$ using \mathbf{y}_{-k} as data, $k = 1, 2, \dots, K$. We compute the cross-validation score for each of the competing models and consider the model with largest score as the best in the model set.

37.11.6 Posterior Predictive Loss

Gelfand and Ghosh [45] recommended a decision theoretic approach for model comparison based on predictive accuracy. The goal of this approach is to obtain good predictions for replicates \mathbf{y}^{rep} of the observed data \mathbf{y} and, at the same time, to be faithful to the observed values. To attain this objective, they proposed a loss function in terms of unobserved replicate \mathbf{y}^{rep} :

$$\mathcal{L}(\mathbf{y}^{\text{rep}}, a; \mathbf{y}) = \mathcal{L}(\mathbf{y}^{\text{rep}}, a) + w \mathcal{L}(\mathbf{y}, a), \tag{37.75}$$

where \mathbf{y}^{rep} has the same first stage distribution as \mathbf{y} and a is an ‘action’ or an estimate for \mathbf{y}^{rep} . In the proposed loss function (37.75), w is constrained to be non-negative and viewed as a relative weight of how important it is for a to be close to \mathbf{y} , compared to \mathbf{y}^{rep} .

Gelfand and Ghosh [45] derived a posterior predictive risk by averaging their proposed loss function (37.75) over the predictive distribution of $y_{i,\text{rep}}$ given \mathbf{y} . The resulting risk is then minimised with respect to the prediction a_i and summed over all observations $i = 1, \dots, d$ to yield the model selection criterion

$$D_w = \sum_{i=1}^d \min_{a_i} \mathbb{E}_{y_i^{\text{rep}} | \mathbf{y}} \left[\mathcal{L}(y_i^{\text{rep}}, a_i) + w \mathcal{L}(y_i, a_i) \right]. \tag{37.76}$$

Provided a loss function $\mathcal{L}(\cdot)$ and weight w , the model with the smallest D_w is favoured. In practice, it can be difficult to carry out the computation in (37.76). If we select the squared error loss function, the criterion becomes

$$D_{w2} = \frac{w}{w+1} \sum_{i=1}^d (y_i - \mathbb{E}(y_{i,\text{rep}} | \mathbf{y}))^2 + \sum_{i=1}^d \text{Var}(y_{i,\text{rep}} | \mathbf{y}). \tag{37.77}$$

Further, by letting $w \rightarrow \infty$, we get

$$D_\infty = \sum_{i=1}^d (y_i - \mathbb{E}(y_i^{\text{rep}} | \mathbf{y}))^2 + \sum_{i=1}^d \text{Var}(y_i^{\text{rep}} | \mathbf{y}). \tag{37.78}$$

Although D_∞ was first proposed by [45, 46] showed that it comes as the limit of D_{w2} . The first term in the D_∞ criterion is the goodness-of-fit term, while the second term in the right-hand side of the above (37.78), i.e. $\sum_{i=1}^d \text{Var}(y_i^{\text{rep}} | \mathbf{y})$, can be viewed as a penalty and will tend to increase in an overfitted model where the prediction variance gets larger with an increasing number of parameters. In general, the ordering of models under D_{w2} agrees with the ordering of D_∞ . Then the model for which D_∞ is the smallest can be chosen while doing model comparison.

We compute the expectation $\mathbb{E}(y_i^{\text{rep}} | \mathbf{y})$ and variance $\text{Var}(y_i^{\text{rep}} | \mathbf{y})$ using MCMC draws. Given an MCMC sample $\{\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(n)}\}$ from $p(\boldsymbol{\theta} | \mathbf{y})$, we simulate $\mathbf{y}^{\text{rep},(m)}$ from $l(\boldsymbol{\theta}^{(m)} | \mathbf{y})$ for each $m = 1, 2, \dots, n$. We compute the quantities as follows:

$$\begin{aligned} \widehat{\mathbb{E}}(y_i^{\text{rep}} | \mathbf{y}) &= \frac{1}{n} \sum_{m=1}^n y_i^{\text{rep},(m)} \\ \widehat{\text{Var}}(y_i^{\text{rep}} | \mathbf{y}) &= \frac{1}{n} \sum_{m=1}^n \left(y_i^{\text{rep},(m)} - \frac{1}{n} \sum_{m=1}^n y_i^{\text{rep},(m)} \right)^2. \end{aligned}$$

37.11.7 Bayes Factor

Let $\boldsymbol{\theta}_i$ denote the vector of parameters under a model \mathcal{M}_i , with prior density $p(\boldsymbol{\theta}_i | \mathcal{M}_i)$ such that $\int p(\boldsymbol{\theta}_i | \mathcal{M}_i) d\boldsymbol{\theta}_i = 1$. Let \mathbf{y} denote the observed data with probability density $l_i(\boldsymbol{\theta}_i | \mathbf{y})$ under \mathcal{M}_i . The posterior density of $\boldsymbol{\theta}_i$ can be expressed as

$$p(\boldsymbol{\theta}_i | \mathbf{y}, \mathcal{M}_i) = \frac{l_i(\boldsymbol{\theta}_i | \mathbf{y}) p(\boldsymbol{\theta}_i | \mathcal{M}_i)}{m(\mathbf{y} | \mathcal{M}_i)} \tag{37.79}$$

where $m(\mathbf{y} | \mathcal{M}_i)$ denotes the marginal likelihood of data \mathbf{y} under \mathcal{M}_i and is computed as

$$m(\mathbf{y} | \mathcal{M}_i) = \int l_i(\boldsymbol{\theta}_i | \mathbf{y}) \pi(\boldsymbol{\theta}_i | \mathcal{M}_i) d\boldsymbol{\theta}_i. \tag{37.80}$$

This marginal likelihood is also viewed as the predictive density of the data under \mathcal{M}_i before the data is actually observed, that is, the probability density of the observed data \mathbf{y} accounting for the prior uncertainty in $\boldsymbol{\theta}_i$ specified by \mathcal{M}_i . Thus the value of the marginal or predictive density has the ability in assessing the merit of the corresponding model. The Bayes factor [see, e.g. 47] of \mathcal{M}_i relative to another model \mathcal{M}_j is defined as the ratio of respective marginal densities

$$\text{BF}(\mathcal{M}_i, \mathcal{M}_j) = \frac{m(\mathbf{y} | \mathcal{M}_i)}{m(\mathbf{y} | \mathcal{M}_j)}. \quad (37.81)$$

Now suppose we want to test

$$\mathcal{M}_0 : \boldsymbol{\theta} \in \Theta_0 \text{ versus } \mathcal{M}_1 : \boldsymbol{\theta} \in \Theta_1. \quad (37.82)$$

Let $\Pr(\mathcal{M}_i)$ be the prior probability of \mathcal{M}_i and $\pi(\boldsymbol{\theta} | \mathcal{M}_i)$ be the prior density of $\boldsymbol{\theta}$ under \mathcal{M}_i such that $\int \pi(\boldsymbol{\theta} | \mathcal{M}_i) d\boldsymbol{\theta} =$

$1, i = 0, 1$. Then the posterior model probability $\Pr(\mathcal{M}_i | \mathbf{y})$ is calculated as

$$\Pr(\mathcal{M}_i | \mathbf{y}) = \frac{m(\mathbf{y} | \mathcal{M}_i) \Pr(\mathcal{M}_i)}{m(\mathbf{y})}, \quad i = 0, 1, \quad (37.83)$$

where $m(\mathbf{y}) = \Pr(\mathcal{M}_0) m(\mathbf{y} | \mathcal{M}_0) + \Pr(\mathcal{M}_1) m(\mathbf{y} | \mathcal{M}_1)$ and $m(\mathbf{y} | \mathcal{M}_0)$ can be calculated as showed in (37.80). For the test, we compute the posterior odds of \mathcal{M}_0 relative to \mathcal{M}_1

$$\begin{aligned} \frac{\Pr(\mathcal{M}_0 | \mathbf{y})}{\Pr(\mathcal{M}_1 | \mathbf{y})} &= \frac{m(\mathbf{y} | \mathcal{M}_0) \Pr(\mathcal{M}_0) / m(\mathbf{y})}{m(\mathbf{y} | \mathcal{M}_1) \Pr(\mathcal{M}_1) / m(\mathbf{y})} = \frac{\Pr(\mathcal{M}_0)}{\Pr(\mathcal{M}_1)} \cdot \frac{m(\mathbf{y} | \mathcal{M}_0)}{m(\mathbf{y} | \mathcal{M}_1)} \\ &= \frac{\Pr(\mathcal{M}_0)}{\Pr(\mathcal{M}_1)} \times \text{BF}(\mathcal{M}_0, \mathcal{M}_1). \end{aligned} \quad (37.84)$$

Consequently, (37.84) gives us

$$\begin{aligned} \text{BF}(\mathcal{M}_0, \mathcal{M}_1) &= \text{Posterior odds}(\mathcal{M}_0, \mathcal{M}_1) \\ &\div \text{Prior odds}(\mathcal{M}_0, \mathcal{M}_1). \end{aligned} \quad (37.85)$$

This shows the importance of Bayes factor in the context of Bayesian testing and model selection. It is to be noted that it does not depend on any prior model probability $\Pr(\mathcal{M}_0)$ or $\Pr(\mathcal{M}_1)$. Higher values of $\text{BF}(\mathcal{M}_0, \mathcal{M}_1)$ suggest stronger evidence in favour of \mathcal{M}_0 . Note that Bayes factor is well defined if the assigned prior distribution on the parameters is proper. However, it can be sensitive to the choice of the prior distribution. Thus it is important to conduct a sensitivity analysis by studying the range of the Bayes factor over a class of possible prior distributions [48].

The computation of marginal likelihood of data is of considerable interest in the Bayesian literature due to its importance in Bayesian model comparison [47]. Marginal likelihood of data is obtained by integrating the likelihood function with respect to the prior distribution, which is also the normalising constant of posterior distribution of the unknown parameter. There is a vast literature on estimating the normalising constant of a given density by Markov chain Monte Carlo (MCMC) methods; more details on this can be found in [49–52]. The existing approaches for computing the marginal likelihood often use MCMC draws, even though, as attractive as it sounds, it comes with its advantages and disadvantages [see the discussion in 53]. In spite of the substantial progress, estimation of marginal likelihoods for hierarchical models is still a difficult problem, especially in high dimensions and in the presence of latent variables.

Laplace Approximation

We start by assuming that the posterior density $p(\boldsymbol{\theta}_i | \mathbf{y}, \mathcal{M}_i)$ is unimodal and has mode at $\tilde{\boldsymbol{\theta}}$. The Laplace approximation of $m(\mathbf{y} | \mathcal{M}_i)$ [6, 54] is obtained as

$$\hat{m}(\mathbf{y} | \mathcal{M}_i) = (2\pi)^{p_i/2} |\tilde{\Sigma}_i|^{1/2} l_i(\tilde{\boldsymbol{\theta}}_i | \mathbf{y}) p(\tilde{\boldsymbol{\theta}}_i | \mathcal{M}_i), \quad (37.86)$$

where $\boldsymbol{\theta}_i$ has dimension p_i and $\tilde{\Sigma}_i = \{-\mathbf{D}^2 \log(l_i(\boldsymbol{\theta}_i | \mathbf{y}) p(\boldsymbol{\theta}_i | \mathcal{M}_i))\}$ denotes the inverse of the negative Hessian matrix. Using this approximation to the $\text{BF}(\mathcal{M}_i, \mathcal{M}_j)$ yields

$$\begin{aligned} \log \text{BF}(\mathcal{M}_i, \mathcal{M}_j) &= \hat{m}(\mathbf{y} | \mathcal{M}_i) - \hat{m}(\mathbf{y} | \mathcal{M}_j) \\ &= l_i(\tilde{\boldsymbol{\theta}}_i | \mathbf{y}) - l_j(\tilde{\boldsymbol{\theta}}_j | \mathbf{y}) + \log p(\tilde{\boldsymbol{\theta}}_i | \mathcal{M}_i) - \log p(\tilde{\boldsymbol{\theta}}_j | \mathcal{M}_j) \\ &\quad + \frac{1}{2}(p_i - p_j) \log(2\pi) + \frac{1}{2}(\log |\tilde{\Sigma}_i| - \log |\tilde{\Sigma}_j|). \end{aligned} \quad (37.87)$$

The above Laplace approximation of $m(\mathbf{y} | \mathcal{M}_i)$ satisfies $m(\mathbf{y} | \mathcal{M}_i) = \hat{m}(\mathbf{y} | \mathcal{M}_i)(1 + O(d^{-1}))$ where d denotes the sample size [54].

Monte Carlo Estimator

Given a sample $\{\boldsymbol{\theta}_i^{(1)}, \boldsymbol{\theta}_i^{(2)}, \dots, \boldsymbol{\theta}_i^{(n)}\}$ from the prior density $p(\boldsymbol{\theta}_i | \mathcal{M}_i)$, we can obtain Monte Carlo estimate of the marginal likelihood $m(\mathbf{y} | \mathcal{M}_i)$ as

$$\hat{m}(\mathbf{y} | \mathcal{M}_i) = \frac{1}{n} \sum_{j=1}^n l_i(\boldsymbol{\theta}_i^{(j)} | \mathcal{M}_i). \quad (37.88)$$

This estimate may be very inaccurate if the sample from the prior density fails to include the values of $\boldsymbol{\theta}$ for which the likelihood is high. Thus a very large sample may be required to improve the accuracy of the estimate.

Importance Sampling Estimator

For a proper density $g(\boldsymbol{\theta}_i | \mathcal{M}_i)$, the following identity holds

$$m(\mathbf{y} | \mathcal{M}_i) = \int l_i(\boldsymbol{\theta}_i | \mathbf{y}) w(\boldsymbol{\theta}_i | \mathcal{M}_i) g(\boldsymbol{\theta}_i | \mathcal{M}_i) d\boldsymbol{\theta}_i, \quad (37.89)$$

where $w(\boldsymbol{\theta}_i | \mathcal{M}_i) = p(\boldsymbol{\theta}_i | \mathcal{M}_i) / g(\boldsymbol{\theta}_i | \mathcal{M}_i)$. The importance density $g(\boldsymbol{\theta}_i | \mathcal{M}_i)$ should be chosen carefully as different

choices of importance density lead to different accuracies for the importance sampling estimator [55]. For the importance sampling estimator to have a smaller variance, the importance density should have thicker tail than the prior density.

Gelfand-Dey Estimator of Marginal Likelihood of Data

Observe that the following identity holds for any proper density $g(\boldsymbol{\theta}_i | \mathcal{M}_i)$,

$$m(\mathbf{y} | \mathcal{M}_i) = \left[\int \frac{g(\boldsymbol{\theta}_i | \mathcal{M}_i) p(\boldsymbol{\theta}_i | \mathbf{y}, \mathcal{M}_i)}{l_i(\boldsymbol{\theta}_i | \mathbf{y}) p(\boldsymbol{\theta}_i | \mathcal{M}_i)} d\boldsymbol{\theta}_i \right]^{-1}. \quad (37.90)$$

Based on this identity (37.90), [49] proposed an estimator for the marginal likelihood $m(\mathbf{y} | \mathcal{M}_i)$

$$\hat{m}_{\text{GD}}(\mathbf{y} | \mathcal{M}_i) = \left[\frac{1}{n} \sum_{j=1}^n \frac{g(\boldsymbol{\theta}_i^{(j)} | \mathcal{M}_i)}{l_i(\boldsymbol{\theta}_i^{(j)} | \mathbf{y}) p(\boldsymbol{\theta}_i^{(j)} | \mathcal{M}_i)} \right]^{-1}, \quad (37.91)$$

where $\{\boldsymbol{\theta}_i^{(1)}, \boldsymbol{\theta}_i^{(2)}, \dots, \boldsymbol{\theta}_i^{(n)}\}$ is a set of MCMC draws from the posterior $\pi(\boldsymbol{\theta}_i | \mathbf{y}, \mathcal{M}_i)$ under the model \mathcal{M}_i . $\hat{m}_{\text{GD}}(\mathbf{y} | \mathcal{M}_i)^{-1}$ is an unbiased and consistent estimator of $m(\mathbf{y} | \mathcal{M}_i)^{-1}$. Even though the estimator is defined for any proper density $g(\boldsymbol{\theta}_i | \mathcal{M}_i)$, shape of $g(\boldsymbol{\theta}_i | \mathcal{M}_i)$ should be similar with thinner tails to the posterior $p(\boldsymbol{\theta}_i | \mathbf{y}, \mathcal{M}_i)$ for Gelfand-Dey estimator to have a smaller variance (like importance sampling). It also satisfies a Gaussian central limit theorem if the tails of $g(\boldsymbol{\theta}_i | \mathcal{M}_i)$ are thin enough, specifically if

$$\int \frac{g(\boldsymbol{\theta}_i | \mathcal{M}_i)^2}{l_i(\boldsymbol{\theta}_i | \mathbf{y}) p(\boldsymbol{\theta}_i | \mathcal{M}_i)} d\boldsymbol{\theta}_i < \infty.$$

In practice, it may be hard to find a density $g(\boldsymbol{\theta}_i | \mathcal{M}_i)$ that has sufficiently thin tails in all direction of the parameter space simultaneously, especially when the posterior has nonelliptical contours. If $g(\boldsymbol{\theta}_i) / \{l_i(\boldsymbol{\theta}_i | \mathbf{y}) p(\boldsymbol{\theta}_i | \mathcal{M}_i)\}$ is bounded above, then the approximation shows less variation over different simulations and the rate of convergence is likely to be practical. In addition, if the posterior density is uniformly bounded away from 0 on every compact subset of the parameter space, then the function $g(\boldsymbol{\theta}_i | \mathcal{M}_i) / \{l_i(\boldsymbol{\theta}_i | \mathbf{y}) p(\boldsymbol{\theta}_i | \mathcal{M}_i)\}$ has posterior moments of all orders [see 56]. Note that choosing $g(\boldsymbol{\theta}_i | \mathcal{M}_i) = p(\boldsymbol{\theta}_i | \mathcal{M}_i)$ gives the harmonic mean estimator:

$$\hat{m}_{\text{HM}}(\mathbf{y} | \mathcal{M}_i) = \left[\frac{1}{n} \sum_{j=1}^n \frac{1}{l_i(\boldsymbol{\theta}_i^{(j)} | \mathbf{y})} \right]^{-1}. \quad (37.92)$$

The Gelfand-Dey estimator is occasionally termed as *reciprocal importance sampling estimator* because of its similarity with importance sampling. For more details please refer to [47, 49–51].

37.11.8 Bayesian Model Averaging

Given a set of models $\{\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_L\}$ with prior model probabilities $\{\text{Pr}(\mathcal{M}_1), \text{Pr}(\mathcal{M}_2), \dots, \text{Pr}(\mathcal{M}_L)\}$, the posterior model probability $\text{Pr}(\mathcal{M}_i | \mathbf{y})$ for model \mathcal{M}_i is calculated as

$$\text{Pr}(\mathcal{M}_i | \mathbf{y}) = \frac{m(\mathbf{y} | \mathcal{M}_i) \text{Pr}(\mathcal{M}_i)}{\sum_{k=1}^L m(\mathbf{y} | \mathcal{M}_k) \text{Pr}(\mathcal{M}_k)}, \quad i = 1, 2, \dots, L. \quad (37.93)$$

where $m(\mathbf{y} | \mathcal{M}_i)$ denotes the marginal likelihood of data \mathbf{y} under \mathcal{M}_i and can be calculated as showed in (37.80). Posterior model probabilities can also be expressed in terms of Bayes factor (defined in (37.81))

$$\text{Pr}(\mathcal{M}_i | \mathbf{y}) = \frac{\text{BF}(\mathcal{M}_i, \mathcal{M}_1) \text{Pr}(\mathcal{M}_i)}{\sum_{k=1}^L \text{BF}(\mathcal{M}_k, \mathcal{M}_1) \text{Pr}(\mathcal{M}_k)}, \quad i = 1, 2, \dots, L. \quad (37.94)$$

The posterior model probabilities can be used to select or compare models, and of course, the models with higher probabilities should be preferred.

Let θ be a quantity of interest, and its marginal posterior density function is expressed as

$$\pi(\theta | \mathbf{y}) = \sum_{k=1}^L \pi(\theta | \mathbf{y}, \mathcal{M}_k) \text{Pr}(\mathcal{M}_k | \mathbf{y}) \quad (37.95)$$

which is a weighted average over all the competing models with weights $\{\text{Pr}(\mathcal{M}_1 | \mathbf{y}), \text{Pr}(\mathcal{M}_2 | \mathbf{y}), \dots, \text{Pr}(\mathcal{M}_L | \mathbf{y})\}$. The posterior means and variances can also be obtained by this model averaging

$$\begin{aligned} \mathbb{E}(\theta | \mathbf{y}) &= \sum_{k=1}^L \hat{\theta}_k \text{Pr}(\mathcal{M}_k | \mathbf{y}) \\ \text{Var}(\theta | \mathbf{y}) &= \sum_{k=1}^L (\text{Var}(\theta | \mathbf{y}, \mathcal{M}_k) + \hat{\theta}_k^2) \\ &\quad \text{Pr}(\mathcal{M}_k | \mathbf{y}) - \mathbb{E}(\theta | \mathbf{y})^2 \end{aligned} \quad (37.96)$$

where $\hat{\theta}_k = \mathbb{E}(\theta | \mathbf{y}, \mathcal{M}_k)$ [57, 58]. For more details on Bayesian model averaging, interested reader can refer to [59, 60].

37.11.9 Bayesian Information Criterion (BIC)

Bayesian information criterion [BIC, 61] is defined as

$$\text{BIC} = -2 \log l(\hat{\boldsymbol{\theta}} | \mathbf{y}) + p \log d, \quad (37.97)$$

where d denotes the sample size and $\hat{\theta}$ denotes the estimate of posterior mean or posterior mode. We calculate BIC for each of the L competing models and choose the one with the minimum BIC value. Note that BIC gives larger correction for bias as sample size increases and thus tends to favour simpler models. BIC is essentially used as an approximation of the marginal likelihood of the data \mathbf{y} and is not an estimate of the predictive accuracy of the model. In this way, it is different than AIC and the other criteria that we will discuss below.

Given a set of models $\{\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_L\}$ and their respective BIC scores $\{\text{BIC}_1, \text{BIC}_2, \dots, \text{BIC}_L\}$, the posterior probability of model \mathcal{M}_i , $i = 1, 2, \dots, L$, can be approximated as [see 62]

$$\Pr(\mathcal{M}_i | \mathbf{y}) \approx \frac{\exp(-\text{BIC}_i/2)}{\sum_{k=1}^L \exp(-\text{BIC}_k/2)}. \quad (37.98)$$

In practice, first we calculate the differences $\Delta_i = \text{BIC}_i - \text{BIC}_{\min}$, where BIC_{\min} is the minimum BIC score among the L competing models. Then we calculate

$$\Pr(\mathcal{M}_i | \mathbf{y}) \approx \frac{\exp(-\Delta_i/2)}{\sum_{k=1}^L \exp(-\Delta_k/2)}. \quad (37.99)$$

Computing the posterior model probabilities in this way helps in avoiding underflow or overflow of large quantities while exponentiating the BIC scores.

37.12 Some Applications of Bayesian Methods in Different Fields of Research

Bayesian concepts have been applied to many areas like process control, reliability and survival analysis, time series, statistical ecology and the like. We will briefly discuss some of these applications in this section. Congdon [63] has prescribed various modelling techniques for different applications. In the case of non-linear and also non-parametric Bayesian models, [3] discuss in details the part V of the book.

37.12.1 Example from Statistical Ecology

Estimation of animal abundances over large regions by integrating multiple data sources is one of the key challenges in statistical ecology. Estimating animal population size is needed to be performed at large geographical scales that will actually be useful for management and conservation. However, reliable estimation of population size is highly resource intensive (e.g. capture-recapture survey) and only possible

at a smaller scale. Cost-effective and less resource-intensive surveys on animal signs are conducted at the larger scale, but these kind of surveys do not yield animal abundance estimate; instead it provides habitat occupancy probability. In an application, [64] have developed a Bayesian smoothing model by using a conditionally autoregressive prior to address the problem. Their approach is briefly described below. Assume that the region under study is divided into p sites and the number of individuals in i^{th} site, denoted by N_i , follows a Poisson distribution with parameter λ_i , $i = 1, 2, \dots, p$. They have modelled λ_i 's as

$$v_i = \log(\lambda_i) = \mathbf{x}'_i \boldsymbol{\beta} + \phi_i, \quad i = 1, 2, \dots, p, \quad (37.100)$$

where $\boldsymbol{\beta}$ is the regression coefficient, \mathbf{x}_i is the vector of covariates and ϕ_i incorporates the spatial association between sites. Here $\boldsymbol{\phi} = (\phi_1, \phi_2, \dots, \phi_p)'$ is given a conditionally autoregressive prior

$$\boldsymbol{\phi} \sim \mathcal{N}_p(\mathbf{0}, \delta^2(D_w - \rho W)^{-1}), \quad (37.101)$$

where $\delta^2 > 0$, $\rho \in (0, 1)$, $W = (w_{ij})$ denotes a $p \times p$ proximity matrix with w_{ij} taking the value 1 if sites i and j are neighbours and 0 otherwise. They set $w_{ii} = 0$ for all i and $D_w = \text{diag}(w_{10}, \dots, w_{p0})$. They use the reliable estimates of λ_i , viz. $\hat{\lambda}_i(ct)$, as responses that are available for the sites $i = 1, 2, \dots, k_1$, ($k_1 < k$) from camera trap data. Estimates of occupancy probabilities $\hat{\psi}_1, \hat{\psi}_2, \dots, \hat{\psi}_p$ are also available for all the sites from count data on animal signs. Now observing that occupancy probability and true latent abundance are related as $\psi_i = \Pr(N_i > 0) = 1 - \exp(-\lambda_i)$, they have transformed the occupancy probability estimates to obtain indicative estimates of abundance $\hat{\lambda}_i(os) = -\log(1 - \hat{\psi}_i)$, $i = 1, 2, \dots, p$. It is also noted that the exact relationship between estimates of occupancy probability ψ_i and estimates of true latent abundance λ_i is very complex in general, and hence estimates $\hat{\lambda}_i(os)$ cannot be perceived as reliable estimates of λ_i . They have used the following notations for further model development: $\hat{v}_i(ct) = \log(\hat{\lambda}_i(ct))$, \mathbf{X} denotes the data on all the covariates including $\hat{\lambda}_i(os)$, $R = \{1, 2, \dots, k_1\}$ denotes the set of sites with reliable data and $S = \{k_1 + 1, k_1 + 2, \dots, p\}$ denotes the set of remaining sites. Consequently, $\mathbf{v} = (\mathbf{v}^R, \mathbf{v}^S)'$, where $\mathbf{v}^R = (v_1, \dots, v_{k_1})'$, $\mathbf{v}^S = (v_{k_1+1}, \dots, v_p)'$, $\hat{\mathbf{v}}^R(ct) = \hat{\mathbf{v}}(ct)$ and $\hat{\boldsymbol{\lambda}}(os) = (\hat{\boldsymbol{\lambda}}^R(os), \hat{\boldsymbol{\lambda}}^S(os))'$, $\mathbf{X} = (\mathbf{X}^R \ \mathbf{X}^S)'$. It is assumed that

$$\hat{\mathbf{v}}(ct) | \mathbf{v}, \sigma^2 \sim \mathcal{N}_{k_1}(\mathbf{v}, \sigma^2 I_{k_1}). \quad (37.102)$$

The prior on \mathbf{v} is then derived from (37.100) and (37.101) as

$$\mathbf{v} | \boldsymbol{\beta}, \rho, \delta^2 \sim \mathcal{N}_p(\mathbf{X}\boldsymbol{\beta}, \delta^2(D_w - \rho W)^{-1}), \quad (37.103)$$

where $\boldsymbol{\beta}$, δ^2 and ρ are hyper-parameters. Then they obtain

$$\hat{\mathbf{v}}(ct) \mid \sigma^2, \boldsymbol{\beta}, \delta^2, \rho \sim \mathcal{N}_{k_1}(\mathbf{X}^T \boldsymbol{\beta}, \sigma^2 \mathbf{I}_{k_1} + \delta^2 \mathbf{Q}_1(\rho)), \quad (37.104)$$

where $\mathbf{Q}_1(\rho)$ is the top $k_1 \times k_1$ square block of $(D_w - \rho W)^{-1}$. In order to obtain the posterior density $\pi(\sigma^2, \boldsymbol{\beta}, \delta^2, \rho \mid \hat{\mathbf{v}}(ct))$, one needs to assign priors on the parameters $\sigma^2, \boldsymbol{\beta}, \delta^2, \rho$. Also note that

$$\begin{pmatrix} \hat{\mathbf{v}}(ct) \\ \mathbf{v}^T \\ \mathbf{v}^S \end{pmatrix} \mid \sigma^2, \boldsymbol{\beta}, \delta^2, \rho \sim \mathcal{N}_{p+k_1} \left(\begin{pmatrix} \mathbf{X}^T \boldsymbol{\beta} \\ \mathbf{X}^T \boldsymbol{\beta} \\ \mathbf{X}^S \boldsymbol{\beta} \end{pmatrix}, \begin{pmatrix} \sigma^2 \mathbf{I}_{k_1} + \delta^2 \mathbf{Q}_1(\rho) & \delta^2 \mathbf{Q}_1(\rho) & \delta^2 \mathbf{Q}_{12}(\rho) \\ \delta^2 \mathbf{Q}_1(\rho) & \delta^2 \mathbf{Q}_1(\rho) & \delta^2 \mathbf{Q}_{12}(\rho) \\ \delta^2 \mathbf{Q}_{21}(\rho) & \delta^2 \mathbf{Q}_{21}(\rho) & \delta^2 \mathbf{Q}_2(\rho) \end{pmatrix} \right), \quad (37.105)$$

where $\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{Q}_{12}$ and $\mathbf{Q}_{21} = \mathbf{Q}'_{12}$ are as in $(D_w - \rho W)^{-1} = \begin{pmatrix} \mathbf{Q}_1(\rho) & \mathbf{Q}_{12}(\rho) \\ \mathbf{Q}_{21}(\rho) & \mathbf{Q}_2(\rho) \end{pmatrix}$. From (37.105), the full posterior of \mathbf{v}^S is obtained as follows:

$$\begin{aligned} \mathbf{v}^S \mid \hat{\mathbf{v}}(ct), \sigma^2, \boldsymbol{\beta}, \delta^2, \rho \sim \mathcal{N}_{p-k_1}(\mathbf{X}^S \boldsymbol{\beta} + \delta^2 \mathbf{Q}_{21}(\rho)(\sigma^2 \mathbf{I}_{k_1} + \delta^2 \mathbf{Q}_1(\rho))^{-1}(\hat{\mathbf{v}}(ct) - \mathbf{X}^T \boldsymbol{\beta}), \\ \delta^2 \mathbf{Q}_2(\rho) - \delta^4 \mathbf{Q}_{21}(\rho)(\sigma^2 \mathbf{I}_{k_1} + \delta^2 \mathbf{Q}_1(\rho))^{-1} \mathbf{Q}_{12}(\rho)). \end{aligned} \quad (37.106)$$

The fully hierarchical Bayes estimate of \mathbf{v}^S and its posterior covariance matrix can be obtained from (37.106). From (37.105),

$$\mathbf{v} \mid \hat{\mathbf{v}}(ct), \sigma^2, \boldsymbol{\beta}, \delta^2, \rho \sim \mathcal{N}_p(\boldsymbol{\mu}(\sigma^2, \boldsymbol{\beta}, \delta^2, \rho), P(\sigma^2, \boldsymbol{\beta}, \delta^2, \rho)), \quad (37.107)$$

where (suppressing the dependence on $\hat{\mathbf{v}}(ct)$)

$$\begin{aligned} \boldsymbol{\mu}(\sigma^2, \boldsymbol{\beta}, \delta^2, \rho) &= \mathbf{X} \boldsymbol{\beta} + \delta^2 (\mathbf{Q}_1(\rho), \mathbf{Q}_{12}(\rho))' (\sigma^2 \mathbf{I}_{k_1} + \delta^2 \mathbf{Q}_1(\rho))^{-1} (\hat{\mathbf{v}}(ct) - \mathbf{X}^T \boldsymbol{\beta}), \\ P(\sigma^2, \boldsymbol{\beta}, \delta^2, \rho) &= \delta^2 \begin{pmatrix} \mathbf{Q}_1(\rho) & \mathbf{Q}_{12}(\rho) \\ \mathbf{Q}_{21}(\rho) & \mathbf{Q}_2(\rho) \end{pmatrix} \\ &\quad - \delta^4 \begin{pmatrix} \mathbf{Q}_1(\rho) \\ \mathbf{Q}_{21}(\rho) \end{pmatrix} (\sigma^2 \mathbf{I}_{k_1} + \delta^2 \mathbf{Q}_1(\rho))^{-1} \begin{pmatrix} \mathbf{Q}_1(\rho) \\ \mathbf{Q}_{21}(\rho) \end{pmatrix}'. \end{aligned} \quad (37.108)$$

Proposition 1 *The hierarchical Bayes estimate of total abundance and its posterior variance are, respectively,*

$$\begin{aligned} \mathbb{E}(A(\boldsymbol{\lambda}) \mid \hat{\mathbf{v}}(ct)) &= \sum_{i=1}^p \mathbb{E}^*(e^{\mu_i + P_{ii}/2}), \text{ and} \\ \text{Var}(A(\boldsymbol{\lambda}) \mid \hat{\mathbf{v}}(ct)) &= \mathbb{E}^* \left(\sum_{i=1}^p e^{2\mu_i + P_{ii}} (e^{P_{ii}} - 1) \right) \\ &\quad + \mathbb{E}^* \left(\sum_{i \neq j} e^{\mu_i + \mu_j + (P_{ii} + P_{jj})/2} (e^{P_{ij}} - 1) \right) + \text{Var}^* \left(\sum_{i=1}^p e^{\mu_i + P_{ii}/2} \right), \end{aligned}$$

where \mathbb{E}^* (and Var^*) denotes expectation (and variance) with respect to the posterior distribution $\pi(\sigma^2, \boldsymbol{\beta}, \delta^2, \rho \mid \hat{\mathbf{v}}(ct))$.

The above result is then used to obtain the posterior estimate of the total abundance. They have also extended their work to perform model comparison by computing the Bayes factors and developed another Bayesian spatial model that alleviates the issue of spatial confounding (i.e. multicollinearity

between spatial covariates and the spatial random effect). They have applied their developed model on tiger population survey data from the Malenad landscape of southwestern India (viz. *Malenad Tiger Landscape (MTL)*). For brevity, we are not going into the results of the application. However, interested readers are suggested to see [64] for details.

37.12.2 Process Control

Statistical process control has been a very interesting topic since the days of Shewhart, the father of Statistical Quality Control. His famous idea of control charts could help organisations for decades to keep control over a process. Later

on, problems crept in the form of less data, for example, for ‘short run’ products. Some statisticians used Bayesian concepts to use some prior ideas about the process parameters. One may refer to [65] for a detailed discussion on various modelling approaches to process control using Bayesian concepts. However, modern-day problems are generally Big Data problems where Bayesian approach can give useful results [66]. Shiau et al. [67], Niverthi and Dey [68], Cheng and Spring [69] and many others have developed Bayesian approaches in measuring process capability of a given process.

37.12.3 Reliability

Several authors have used Bayesian methodology to reliability theory. Reliability domain can broadly be classified as hardware and software. Research on hardware reliability theory has been continuing over several decades, whereas software reliability concepts and models have its origin in 1972 [70]. Some of the latest applications of Bayesian methodology in software reliability aspects can be seen in [71] and [72]. The authors have optimised the time for software release using Bayesian concepts. Various other authors have found optimum release time of software using non-Bayesian methods.

In their latest article [72], the authors propounded a new direction to develop software reliability models. Reliability of software depends very much on the remaining number of bugs in the software. However, it has been observed that even if bugs are there in the software, but if the data does not go through the path where the bug exists, then the software does not create a problem in getting the right answer. This necessitates the development of the new concept of size of a bug, where the size here basically means the number of input data that may eventually traverse through the path where the bug exists. So, even if several bugs exist in the software, but all of them are in such paths which are rarely traversed by the data sets to be used by the users, the software is expected to show high reliability. This idea was modelled through the concept of size-biased sampling, originally developed in [73]. On the hardware reliability front, Bayesian methods have been used by several authors; see, for example, [74, 75], etc.

37.13 Discussion

Although Bayesian analysis provides a natural and comprehensive approach of combining prior information with data, within a solid decision theoretical framework, it has certain issues and criticisms which stem from the paradigm.

37.13.1 Pitfalls in the Choice of Prior

As we have seen above, there are many ways to construct a prior; there is no uniformly correct way to select it. As it requires skills to mathematically formulate subjective prior information, it may generate misleading results if not proceeded carefully. Scientists who are experts in their own field but not skilled enough with the practice of Bayesian analysis can construct a prior distribution which is inconsistent and imprecise [76]. Often posterior distributions are heavily influenced by the priors which makes the results misleading and raises questions on the selection of the prior. In the absence of sufficient amount of data, the choice of prior can lead to wrong conclusions in the case of limited prior information. It is to be noted that the majority of the concerns encountered during assessment of a Bayesian analysis arise from this issue regarding the choice of prior distributions. The construction of informative priors is productive and suggested in general. If possible, involvement of subject matter experts (i.e. biologists, industry, conservationist, etc.) should be ensured in the process of construction of the priors. This helps in correctly interpreting and defending the results.

37.13.2 Computationally Intensive

It becomes extremely computationally challenging to conduct a Bayesian analysis in complex models and in models with a large number of parameters. The computational challenges come in with both mathematical skills (requiring the knowledge of probability and algebra) and computational time (requiring a high performance computer with higher RAM, cores and processing power). The posterior distributions often do not have a proper and/or standard form necessitating simulation (such as Monte Carlo or MCMC, etc.). These simulations provide a little different answers if different random seeds are used, i.e. for a particular posterior distribution, simulation-based estimates can vary for different random number generator used in the analyses. For more comprehensive and thorough treatments on the advantages and disadvantages of Bayesian analysis, see [37, 77–82].

37.13.3 Bayesian Marginalisation Paradox

Now it is well-known to the Bayesian community that apparent inconsistency may arise in the Bayesian inference due to the use of improper priors. The marginalisation paradox is a particular type of such inconsistency which arises when there are different approaches to compute the same marginal posterior [83, 84], resulting in incompatible densities by using improper priors. This inconsistency does not arise by using proper priors.

If $p(\mathbf{x} | \boldsymbol{\theta})$ denotes the density function for a certain statistical model, $\pi(\boldsymbol{\theta})$ denotes the assumed improper prior and $\pi(\boldsymbol{\theta} | \mathbf{x})$ denotes the posterior distribution of the parameter $\boldsymbol{\theta}$ given the data \mathbf{x} , the posterior $\pi(\boldsymbol{\theta} | \mathbf{x})$ is computed by using the Bayes theorem. Now suppose that $\boldsymbol{\theta} = (\theta_1, \theta_2)$ and $\mathbf{x} = (x_1, x_2)$. We also assume that the marginal density $p(x_2 | \boldsymbol{\theta})$ depends on $\boldsymbol{\theta}$ only through θ_2 and that the marginal posterior $\pi(\theta_2 | \mathbf{x})$ depends on \mathbf{x} only through x_2 . Hence we should have the posterior of θ_2 given x_2 , $\pi(\theta_2 | x_2) = p(x_2 | \theta_2) \pi(\theta_2)$ when density function of x_2 is denoted by $p(x_2 | \theta_2)$ with some prior density $\pi(\theta_2)$. This marginal density does not satisfy the Bayes theorem, although it should have. One could integrate $\pi(\theta_1, \theta_2 | x_1, x_2) \propto p(x_1, x_2 | \theta_1, \theta_2) \pi(\theta_1, \theta_2)$ with respect to θ_1 to obtain the marginal posterior density $\pi(\theta_2 | x_2)$. As an alternative approach, one could first integrate $p(x_1, x_2 | \theta_1, \theta_2) \pi(\theta_1, \theta_2)$ with respect to x_1 to obtain $p(x_2 | \theta_2)$ and then obtain $p(x_2 | \theta_2) \pi(\theta_2)$. It turns out $\pi(\theta_2 | x_2)$ obtained by the former approach is not always proportional to $p(x_2 | \theta_2) \pi(\theta_2)$ that is obtained in the latter approach. There are numerous examples of such inconsistency that arise in ordinary problems, e.g. the change point problem and the problem of discrimination parameter for two populations [84]. This paradox is important as it shows certain inconsistency which raises concerns regarding the use of improper priors in objective Bayes. Recently, under some conditions, it has been shown that the marginalisation paradox can be resolved by considering the improper inferences as probability limits [85].

37.13.4 Bayesian Discrepant Posterior Phenomenon

The posterior distribution combines the information obtained from the prior and the likelihood to provide estimates of a parameter – hence it can be expected to give an improved estimate. For example, the posterior estimate of the mean parameter in a Gaussian conjugate model (with unknown mean and known variance) is a weighted average of the maximum likelihood estimate and the prior mean. Consequently the posterior estimate lies between the estimates based on the likelihood and the prior in this model. But in a model with multivariate parameters and an informative prior, we may get counterintuitive inference on the parameter of interest based on its marginal posterior density (which is obtained by marginalising the joint posterior density), i.e. it is possible to obtain a point estimate of the parameter of interest based on its marginal posterior density which can be regarded as extreme under the prior and the posterior. This is called the discrepant posterior phenomenon (DPP) [86, 87].

For example, in a binomial clinical trial conducted by Johnson & Johnson (J&J) Inc., both the prior density and the

likelihood of the improvement $\delta = p_1 - p_0$, where p_0 denotes the control success rate and p_1 denotes the treatment success rate, had its mean, median and mode around 0.1. However, it was found that the posterior of δ has its mean, median and mode around 0.2 from several competing Bayesian models [86, see Table 3]. In Fig. 4 of [86] and Fig. 2(b) of [87], it can also be seen that the marginal posterior density of δ peaks near the region that is less likely under the marginal prior and the marginal likelihood of δ . The readers are advised to refer to [88] who provides a comprehensive discussion of the DPP. They recommend (1) to have uncorrelated dimensions for the parameters in both the prior and likelihood; (2) to do re-parameterisation for prior-likelihood curvature alignment such that the prior and likelihood is not highly skewed and homogeneous variance across dimensions is desired; and (3) in case the different dimensions of the parameter are correlated in both the prior and the likelihood, it should be made sure that the correlation patterns are the same between the prior and the likelihood which could alleviate or prevent DPP. When informative prior is needed to be used, their discussion can serve as a precaution and guidance for the Bayesian practitioners for prior constructions to mitigate the effect of DPP.

37.13.5 Lack of Inference Results

Often the Bayesian practitioners encounter a situation where an approximation of intractable integrals is needed to obtain inferences. Variational Bayesian methods (VBM) provide suites of techniques for approximating such integrals. VBMs are primarily used to provide an analytical approximation to the posterior probability of the unobserved variables (i.e. the unknown parameters and the latent variables). VBM can be seen as a quicker alternative to MCMC methods providing locally optimal exact analytic solution to posterior approximation. However, the derivation of the set of equations that are used in the iterative updates of the parameters requires more intensive mathematical work [89, 90].

Variational Bayes can be seen as an extension of the EM (expectation-maximisation) algorithm based on dependent set of equations that cannot be solved analytically – both methods require finding a maximum a posteriori (MAP) estimate of each parameter and approximating the posterior density. But unlike EM algorithm, both types of unobserved variables (i.e. parameters and latent variables) are treated as the same, i.e. as random variables. Estimates for the variables can then be derived in the standard Bayesian ways, e.g. calculating the mean of the distribution to get a single point estimate or deriving a credible interval, highest density region, etc. [91].

References

1. Uusitalo, L.: Advantages and challenges of bayesian networks in environmental modelling. *Ecol. Model.* **203**(3–4), 312–318 (2007)
2. Christensen, R.: *Plane Answers to Complex Questions: The Theory of Linear Models*, 4th edn. Springer-Verlag, New York (2011)
3. Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A., Rubin, D.B.: *Bayesian Data Analysis*, 3rd edn. CRC press, Taylor & Francis Group, Boca Raton, FL (2014)
4. de Finetti, B.: *Theory of Probability*, vol. 1. Wiley, New York (1974)
5. de Finetti, B.: *Theory of Probability*, vol. 2. Wiley, New York (1975)
6. Tierney, L., Kadane, J.B.: Accurate approximations for posterior moments and marginal densities. *J. Am. Stat. Assoc.* **81**(393), 82–86 (1986)
7. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. *J. R. Stat. Soc. Ser. B (Methodol.)* **39**(1), 1–22 (1977)
8. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**(6), 1087–1092 (1953)
9. Hastings, W.K.: Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**(1), 97–109 (1970)
10. Devroye, L.: Non-uniform random variate generation. *Handbooks Oper. Res. Manag. Sci.* **13**, 83–121 (2006)
11. Ripley, B.D.: *Stochastic simulation*, vol. 316. John Wiley & Sons, New York (2009)
12. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721–741. IEEE (1984)
13. Geyer, C.J.: On the convergence of Monte Carlo maximum likelihood calculations. *J. R. Stat. Soc. Ser. B (Methodol.)* **56**(1), 261–274 (1994)
14. Geyer, C.J.: Likelihood inference for spatial point processes. In: Barndorff-Nielsen, O.E., Kendall, W.S., van Lieshout, M.N.M. (eds.) *Stochastic Geometry: Likelihood and Computation*, pp. 78–140. Chapman & Hall/CRC, Boca Raton, FL (1999)
15. Geyer, C.J., Thompson, E.A.: Constrained monte carlo maximum likelihood for dependent data. *J. R. Stat. Soc. Ser. B (Methodol.)* **54**(3), 657–683 (1992)
16. Geyer, C.J., Thompson, E.A.: Annealing markov chain monte carlo with applications to ancestral inference. *J. Am. Stat. Assoc.* **90**(431), 909–920 (1995)
17. Gilks, W.R., Richardson, S., Spiegelhalter, D.: *Markov chain Monte Carlo in practice*. CRC Press, Boca Raton, FL (1995)
18. Meyn, S.P., Tweedie, R.L.: *Markov chains and stochastic stability*. Springer-Verlag, London (2012)
19. Kass, R.E., Carlin, B.P., Gelman, A., Neal, R.M.: Markov chain monte carlo in practice: a roundtable discussion. *Am. Stat.* **52**(2), 93–100 (1998)
20. Geyer, C.J.: Practical Markov chain Monte Carlo. *Stat. Sci.* **7**(4), 473–483 (1992)
21. Glynn, P.W., Whitt, W.: Estimating the asymptotic variance with batch means. *Oper. Res. Lett.* **10**(8), 431–435 (1991)
22. Glynn, P.W., Iglehart, D.L.: Simulation output analysis using standardized time series. *Math. Oper. Res.* **15**(1), 1–16 (1990)
23. Jones, G.L., Haran, M., Caffo, B.S., Neath, R.: Fixed-width Output Analysis for Markov Chain Monte Carlo. *J. Am. Stat. Assoc.* **101**(476), 1537–1547 (2006)
24. Flegal, J.M., Jones, G.L.: Batch means and spectral variance estimators in Markov chain Monte Carlo. *Ann. Stat.* **38**(2), 1034–1070 (2010)
25. Robert, C.P., Casella, G.: *Monte Carlo Statistical Methods*. Springer Texts in Statistics, 2nd edn. Springer-Verlag, New York (2004)
26. Brooks, S., Gelman, A., Jones, G., Meng, X.-L.: *Handbook of Markov chain Monte Carlo*. CRC Press, Boca Raton, FL (2011)
27. Roberts, G.O., Rosenthal, J.S.: General state space Markov chains and MCMC algorithms. *Probab. Surv.* **1**, 20–71 (2004)
28. Tierney, L.: Markov chains for exploring posterior distributions. *Ann. Stat.* **22**(4), 1701–1728 (1994)
29. Gelman, A., Roberts, G.O., Gilks, W.R., et al.: Efficient Metropolis jumping rules. In: Bernardo, J.M., Berger, J.O., Dawid, A.P., Smith, A.F.M. (eds.) *Bayesian Statistics*, vol. 5, pp. 599–608. Oxford University Press (1996)
30. Roberts, G.O., Gelman, A., Gilks, W.R.: Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Probab.* **7**(1), 110–120 (1997)
31. Gelfand, A.E., Smith, A.F.M.: Sampling-based approaches to calculating marginal densities. *J. Am. Stat. Assoc.* **85**(410), 398–409 (1990)
32. Roberts, G.O., Smith, A.F.M.: Simple conditions for the convergence of the Gibbs sampler and Metropolis-Hastings algorithms. *Stoch. Process. Appl.* **49**(2), 207–216 (1994)
33. Besag, J.: Spatial interaction and the statistical analysis of lattice systems. *J. R. Stat. Soc. Ser. B (Methodol.)* **36**, 192–236 (1974)
34. Brook, D.: On the distinction between the conditional probability and the joint probability approaches in the specification of nearest-neighbour systems. *Biometrika* **51**(3–4), 481–483 (1964)
35. Neal, R.M.: Slice sampling. *Ann. Stat.* **31**(3), 705–767 (2003)
36. Green, P.J.: Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination. *Biometrika* **82**(4), 711–732 (1995)
37. Carlin, B.P., Louis, T.A.: *Bayesian Methods for Data Analysis*. CRC Press (2008)
38. Geweke, J.: *Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments*, vol. 196. Federal Reserve Bank of Minneapolis, Research Department Minneapolis, MN (1991)
39. Akaike, H.: Information theory and an extension of the maximum likelihood principle. In: Petrov, B.N., Csaki, F. (eds.) *Proceedings of the Second International Symposium on Information Theory*, pp. 267–281. Akadémiai Kiadó, Budapest (1973). Reprinted in Kotz, S. (ed.) *Breakthroughs in Statistics*, pp. 610–624. Springer, New York (1992)
40. Stone, M.: An asymptotic equivalence of choice of model by cross-validation and Akaike’s criterion. *J. R. Stat. Soc. Ser. B (Methodol.)* **39**(1), 44–47 (1977)
41. Spiegelhalter, D.J., Best, N.G., Carlin, B.P., van der Linde, A.: Bayesian measures of model complexity and fit. *J. R. Stat. Soc. Ser. B (Methodol.)* **64**(4), 583–639 (2002)
42. Gelman, A., Hwang, J., Vehtari, A.: Understanding predictive information criteria for Bayesian models. *Stat. Comput.* **24**(6), 997–1016 (2014)
43. Shibata, R.: Statistical aspects of model selection. In: *From Data to Model*, pp. 215–240. Springer (1989)
44. Watanabe, S.: Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *J. Mach. Learn. Res.* **11**(12), 3571–3594 (2010)
45. Gelfand, A.E., Ghosh, S.K.: Model choice: a minimum posterior predictive loss approach. *Biometrika* **85**(1), 1–11 (1998)
46. Laud, P.W., Ibrahim, J.G.: Predictive model selection. *J. R. Stat. Soc. Ser. B (Methodol.)* **57**(1), 247–262 (1995)
47. Kass, R.E., Raftery, A.E.: Bayes factors. *J. Am. Stat. Assoc.* **90**(430), 773–795 (1995)
48. Ghosh, J.K., Delampady, M., Samanta, T.: *An Introduction to Bayesian Analysis, Theory and Methods*. Springer Texts in Statistics. Springer, New York (2006)
49. Gelfand, A.E., Dey, D.K.: Bayesian model choice: asymptotics and exact calculations. *J. R. Stat. Soc. Ser. B (Methodol.)* **56**(3), 501–514 (1994)

50. Newton, M.A., Raftery, A.E.: Approximate Bayesian inference with the weighted likelihood bootstrap. *J. R. Stat. Soc. Ser. B (Methodol.)* **56**(1), 3–48 (1994)
51. Chib, S.: Marginal likelihood from the Gibbs output. *J. Am. Stat. Assoc.* **90**(432), 1313–1321 (1995)
52. Chib, S., Jeliazkov, I.: Marginal likelihood from the Metropolis–Hastings output. *J. Am. Stat. Assoc.* **96**(453), 270–281 (2001)
53. Chan, J.J.C., Eisenstat, E.: Marginal likelihood estimation with the Cross-Entropy method. *Econometric Rev.* **34**(3), 256–285 (2015)
54. Kass, R.E., Tierney, L., Kadane, J.B.: Laplace’s method in Bayesian analysis. *Contemp. Math.* **115**, 89–99 (1991)
55. Geweke, J.: Bayesian inference in econometric models using Monte Carlo integration. *Econometrica* **57**(6), 1317–1339 (1989)
56. Geweke, J.: Using simulation methods for Bayesian econometric models: inference, development, and communication. *Econometric Rev.* **18**(1), 1–73 (1999)
57. Raftery, A.E.: Bayesian model selection in structural equation models. *Sage Focus Editions* **154**, 163–163 (1993)
58. Draper, D.: Assessment and propagation of model uncertainty. *J. R. Stat. Soc. Ser. B (Methodol.)* **57**(1), 45–70 (1995)
59. Hoeting, J.A., Madigan, D., Raftery, A.E., Volinsky, C.T.: Bayesian model averaging: a tutorial. *Stat. Sci.* **14**(4), 382–401 (1999)
60. Fragoso, T.M., Bertoli, W., Louzada, F.: Bayesian model averaging: A systematic review and conceptual classification. *Int. Stat. Rev.* **86**(1), 1–28 (2018)
61. Schwarz, G.E.: Estimating the dimension of a model. *Ann. Stat.* **6**(2), 461–464 (1978)
62. Neath, A.A., Cavanaugh, J.E.: The Bayesian information criterion: background, derivation, and applications. *Wiley Interdiscip. Rev. Comput. Stat.* **4**(2), 199–203 (2012)
63. Congdon, P.: Applied bayesian modelling, vol. 595. John Wiley & Sons, Chichester, UK (2014)
64. Dey, S., Delampady, M., Parameshwaran, R., Kumar, N.S., Srivathsa, A., Karanth, K.U.: Bayesian methods for estimating animal abundance at large spatial scales using data from multiple sources. *J. Agric. Biol. Environ. Stat.* **22**(2), 111–139 (2017)
65. Colosimo, B.M., Del Castillo, E.: Bayesian process monitoring, control and optimization. CRC Press, Boca Raton, FL (2006)
66. Yau, C., Campbell, K.: Bayesian statistical learning for big data biology. *Biophys. Rev.* **11**(1), 95–102 (2019)
67. Shiau, J.J.H., Chiang, C.-T., Hung, H.-N.: A Bayesian procedure for process capability assessment. *Qual. Reliab. Eng. Int.* **15**(5), 369–378 (1999)
68. Niverthi, M., Dey, D.K.: Multivariate process capability a Bayesian perspective. *Commun. Stat. Simul. Comput.* **29**(2), 667–687 (2000)
69. Cheng, S.-W., Spring, F.A.: Assessing process capability: a Bayesian approach. *IIE Trans.* **21**(1), 97–98 (1989)
70. Jelinski, Z., Moranda, P.: Software reliability research. In: *Statistical computer performance evaluation*, pp. 465–484. Elsevier (1972)
71. Chakraborty, A.K., Basak, G.K., Das, S.: Bayesian optimum stopping rule for software release. *OPSEARCH* **56**(1), 242–260 (2019)
72. Dey, S., Chakraborty, A.K.: Estimating software reliability using size-biased modelling. *arXiv preprint arXiv:2202.08107* (2022)
73. Patil, G.P., Rao, C.R.: Weighted distributions and size-biased sampling with applications to wildlife populations and human families. *Biometrics.* **34**(2), 79–189 (1978)
74. Martz, H.F., Waller, R.: *Bayesian Reliability Analysis*, vol. 704. John Wiley & Sons, New York (1982)
75. Coolen, F.P.A.: On bayesian reliability analysis with informative priors and censoring. *Reliab. Eng. Syst. Saf.* **53**(1), 91–98 (1996)
76. Walters, C., Ludwig, D.: Calculation of Bayes posterior probability distributions for key population parameters. *Can. J. Fish. Aquat. Sci.* **51**(3), 713–722 (1994)
77. Berger, J.O.: *Statistical Decision Theory and Bayesian Analysis*. Springer Series in Statistics, 2nd edn. Springer-Verlag, New York (1985)
78. Berger, J.O., Wolpert, R.L.: *The likelihood principle*. Lecture notes – monograph series, 2nd edn. Institute of Mathematical Statistics, Hayward, California (1988)
79. Robert, C.P.: *The Bayesian Choice, From Decision–Theoretic Foundations to Computational Implementation*. Springer Texts in Statistics, 2nd edn. Springer, New York (2007)
80. Bernardo, J.M., Smith, A.F.M.: *Bayesian theory*, vol. 405. John Wiley & Sons, Chichester, UK (2009)
81. Christensen, R., Johnson, W., Branscum, A., Hanson, T.E.: *Bayesian ideas and data analysis: an introduction for scientists and statisticians*. CRC Press, Boca Raton, FL (2011)
82. Kadane, J.B.: *Principles of uncertainty*. CRC Press, Boca Raton, FL (2011)
83. Stone, M., Dawid, A.P.: Un-Bayesian implications of improper bayes inference in routine statistical problems. *Biometrika* **59**(2), 369–375 (1972)
84. Dawid, A.P., Stone, M., Zidek, J.V.: Marginalization paradoxes in Bayesian and structural inference. *J. R. Stat. Soc. Ser. B (Methodol.)* **35**(2), 189–213 (1973)
85. Wallstrom, T.C.: The marginalization paradox and probability limits. In: J.M. Bernardo et al. (eds.) *Bayesian statistics*, vol. 8, pp. 669–674. Oxford University Press, New York (2007)
86. Xie, M., Liu, R.Y., Damaraju, C.V., Olson, W.H.: Incorporating external information in analyses of clinical trials with binary outcomes. *Ann. Appl. Stat.* **7**(1), 342–368 (2013)
87. Xie, M.-g., Singh, K.: Confidence distribution, the frequentist distribution estimator of a parameter: a review. *Int. Stat. Rev.* **81**(1), 3–39 (2013)
88. Chen, Y., Gong, R., Xie, M.-G.: Geometric conditions for the discrepant posterior phenomenon and connections to Simpson’s paradox. *arXiv preprint arXiv:2001.08336* (2020)
89. Šmídl, V., Quinn, A.: *The variational bayes method in signal processing*. Springer-Verlag Berlin Heidelberg, Germany (2006)
90. Kingma, D.P., Welling, M.: Auto-encoding Variational Bayes. *arXiv preprint arXiv:1312.6114* (2013)
91. Tzikas, D.G., Likas, A.C., Galatsanos, N.P.: The variational approximation for Bayesian inference. *IEEE Signal Process. Mag.* **25**(6), 131–146 (2008)



Dr. Ashis Kumar Chakraborty is a B.stat and M.stat from Indian Statistical Institute (ISI) Kolkata and Ph.D from Indian Institute of Science, Bangalore. His research interest is in process control, reliability, particularly software reliability and the interface area of Statistics and Machine learning. He has published three books and another book is in press with Taylor & Francis. He also contributed chapters in many other books. He has published more than 70 articles in various national and international journals. He is an elected member of the council of Operations Research Society of India (ORSI). He also serves as associate editor of two journals.



Soumen Dey received his Ph.D in Statistics from Indian Statistical Institute in 2018. He worked as a visiting scientist in the Kolkata centre of the same institute in 2019. He works now in Norwegian University of Life sciences (NMBU) as a postdoctoral researcher, focusing on research problems in spatial modelling and Bayesian model checking.



Aleena Chanda completed her Bachelors degree in Statistics from St. Xavier's Kolkata in the year 2016 and her Masters degree in Statistics from the University of Calcutta in the year 2018. She worked as a Visiting Scholar in the Indian Statistical Institute in the SQC and OR Unit under the supervision of Dr. Ashis Kumar Chakraborty in the year 2018–2019. She is currently a PhD student in the University of Nebraska-Lincoln.



Poulami Chakraborty has done her Bachelor's in Statistics (2013–2016) as well as M.Sc (2016–2018) from University of Calcutta. She worked at the Indian Statistical Institute, Kolkata in the SQC and OR unit (2018–2019). She is currently pursuing her PhD in Statistics in the Ohio State University (2019-present).

Statistical Computing and Data Mining



Kwok-Leung Tsui, Victoria Chen, Wei Jiang, Fangfang Yang, and Chen Kan

Contents

38.1	The KDD Process	798
38.2	Handling Data	799
38.2.1	Databases and Data Warehousing	799
38.2.2	Data Preparation	800
38.3	Data Mining (DM) Models and Algorithms	800
38.3.1	Supervised Learning	801
38.3.2	Unsupervised Learning	806
38.3.3	Software	809
38.4	DM Research and Applications	810
38.4.1	Activity Monitoring	810
38.4.2	Mahalanobis–Taguchi System	811
38.4.3	Manufacturing Process Modeling	811
38.4.4	Object Detection	813
38.4.5	Surveillance of Public Health Systems	813
38.5	Concluding Remarks	813
References	814

introduction defines and provides a general background to data mining knowledge discovery in databases, following by an outline of the entire process in the second part. The third part presents data handling issues, including databases and preparation of the data for analysis. The fourth part, as the core of the chapter, describes popular data mining methods, separated as supervised versus unsupervised learning. Supervised learning methods are described in the context of both regression and classification, beginning with the simplest case of linear models, then presenting more complex modeling with trees, neural networks, and support vector machines, and concluding with some methods only for classification. Unsupervised learning methods are described under two categories: association rules and clustering. The fifth part presents past and current research projects, involving both industrial and business applications. Finally, the last part provides a brief discussion on remaining problems and future trends.

Abstract

In this chapter, we provide a review of the knowledge discovery process, including data handling, data mining methods and software, and current research activities. The

Keywords

Data mining · Linear discriminant analysis · Association rule · Unsupervised learning · Statistical process control

K.-L. Tsui (✉)

Grado Department of Industrial and Systems Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA
e-mail: klttsui@vt.edu

V. Chen · C. Kan

Department of Industrial, Manufacturing, & Systems Engineering, University of Texas at Arlington, Arlington, TX, USA
e-mail: vchen@uta.edu; chen.kan@uta.edu

W. Jiang

Antai College of Economics and Management, Shanghai Jiao Tong University, Shanghai, China
e-mail: jiangwei@sjtu.edu.cn

F. Yang

School of Intelligent Systems Engineering Sun Yat-sen University, Guangdong, China
e-mail: yangff7@mail.sysu.edu.cn

Data mining (DM) is the process of exploration and analysis, by automatic or semiautomatic means, of large quantities of data to discover meaningful patterns and rules [1]. Statistical DM is exploratory data analysis with little or no human interaction using computationally feasible techniques, i.e., the attempt to find *unknown interesting* structure [2]. Knowledge discovery in databases (KDD) is a multidisciplinary research field for nontrivial extraction of implicit, previously unknown, and potentially useful knowledge from data [3]. Although some treat DM and KDD equivalently, they can be distinguished as follows. The KDD process employs DM methods (algorithms) to extract knowledge according to the specifications of measures and thresholds, using a database

along with any necessary preprocessing or transformations. DM is a step in the KDD process consisting of particular algorithms (methods) that, under some acceptable objective, produce particular patterns or knowledge over the data. The two primary fields that develop DM methods are statistics and computer science. Statisticians support DM by mathematical theory and statistical methods while computer scientists develop computational algorithms and relevant software [4]. Prerequisites for DM include: (1) advanced computer technology (large CPU, parallel architecture, etc.) to allow fast access to large quantities of data and enable computationally intensive algorithms and statistical methods and (2) knowledge of the business or subject matter to formulate the important business questions and interpret the discovered knowledge.

With competition increasing, DM and KDD have become critical for companies to retain customers and ensure profitable growth. Although most companies are able to collect vast amounts of business data, they are often unable to leverage this data effectively to gain new knowledge and insights. DM is the process of applying sophisticated analytical and computational techniques to discover exploitable patterns in complex data. In many cases, the process of DM results in actionable knowledge and insights. Examples of DM applications include fraud detection, risk assessment, customer relationship management, cross-selling, insurance, banking, retail, etc.

While many of these applications involve customer relationship management in the service industry, a potentially fruitful area is performance improvement and cost reduction through DM in industrial and manufacturing systems. For example, in the fast-growing and highly competitive electronics industry, total revenue worldwide in 2003 was estimated to be \$900 billion, and the growth rate is estimated at 8% per year (www.selectron.com). However, economies of scale, purchasing power, and global competition are making the business such that one must either be a big player or serve a niche market. Today, extremely short life cycles and constantly declining prices are pressuring the electronics industry to manufacture their products with high quality, high yield, and low production cost.

To be successful, industry will require improvements at all phases of manufacturing. Figure 38.1 illustrates the three primary phases: design, ramp-up, and production. In the production phase, maintenance of a high-performance level via improved system diagnosis is needed. In the ramp-up phase, reduction in new product development time is sought by achieving the required performance as quickly as possible. Market demands have been forcing reduced development time for new product and production system design. For example, in the computer industry, a product's life cycle has been shortened to 2–3 years recently, compared to a life cycle of 3–5 years a few years ago. As a result, there are a number of new concepts in the area of production systems,

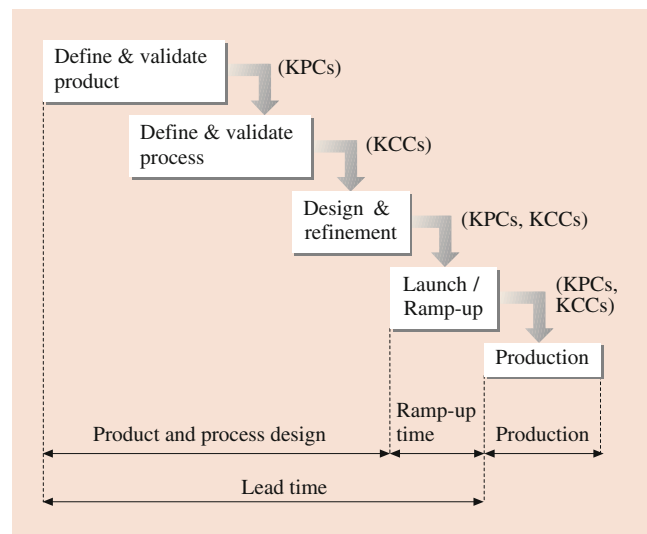


Fig. 38.1 Manufacturing system development phases. KPCs key product characteristics, KCCs key control characteristics

such as flexible and reconfigurable manufacturing systems. Thus, in the design phase, improved system performance integrated at both the ramp-up and production phases is desired. Some of the most critical factors and barriers in the competitive development of modern manufacturing systems lie in the largely uncharted area of predicting system performance during the design phase [5, 6]. Consequently, current systems necessitate that a large number of design/engineering changes be made after the system has been designed.

At all phases, system performance depends on many manufacturing process stages and hundreds or thousands of variables whose interactions are not well understood. For example, in the multistage printed circuit board (PCB) industry, the stages include process operations such as paste printing, chip placement, and wave soldering, and also include test operations such as optical inspection, vision inspection, and functional test. Due to advancements in information technology, sophisticated software and hardware technologies are available to record and process huge amounts of daily data in these process and testing stages. This makes it possible to extract important and useful information to improve process and product performance through DM and quality improvement technologies.

38.1 The KDD Process

The KDD process consists of four main steps:

1. Determination of business objectives
2. Data preparation
 - (a) Create target datasets
 - (b) Data quality, cleaning, and preprocessing
 - (c) Data reduction and projection

3. Data mining
 - (a) Identify DM tasks
 - (b) Apply DM tools
4. Consolidation and application
 - (a) Consolidate discovered knowledge
 - (b) Implement in business decisions

As an example of formulating business objectives, consider a telecommunications company. It is critically important to identify those customer traits that retain profitable customers and predict fraudulent behavior, credit risks, and customer churn. This knowledge may be used to improve programs in target marketing, marketing channel management, micromarketing, and cross-selling. Finally, continually updating this knowledge will enable the company to meet the challenges of new product development effectively in the future. Steps 2–4 are illustrated in Figs. 38.2, 38.3, and 38.4. Approximately 20–25% of effort is spent on determining business objectives, 50–60% of effort is spent on data preparation, 10–15% is spent on DM, and about 10% is spent on consolidation/application.

38.2 Handling Data

The largest percentage effort of the KDD process is spent on processing and preparing the data. In this section, common forms of data storage and tools for accessing the data are described, and the important issues in data preparation are discussed.

38.2.1 Databases and Data Warehousing

A *relational database* system contains one or more objects called tables. The data or information for the database are stored in these tables. Tables are uniquely identified by their names and are comprised of columns and rows. Columns contain the column name, data type, and any other attributes for the column. Rows contain the records or data for the columns. The structured query language (SQL) is the communication tool for relational database management systems. SQL statements are used to perform tasks such as updating data in a database, or retrieving data from a database. Some

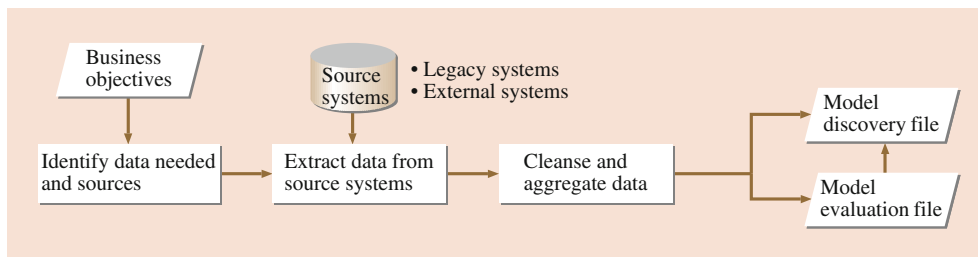


Fig. 38.2 Data preparation flow chart

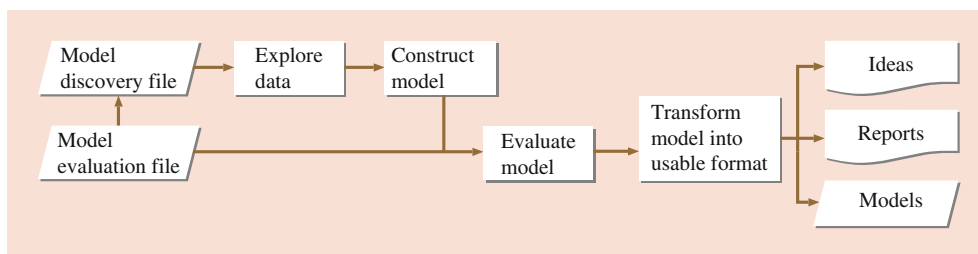


Fig. 38.3 Data mining flow chart

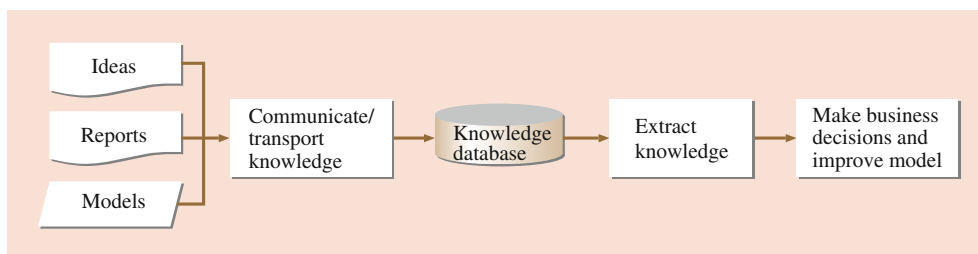


Fig. 38.4 Consolidation and application flow chart

common relational database management systems that use SQL are: Oracle, Sybase, Microsoft SQL Server, Access, and Ingres. Standard SQL commands, such as *select*, *insert*, *update*, *delete*, *create*, and *drop*, can be used to accomplish almost everything that one needs to do with a database.

A *data warehouse* holds local databases assembled in a central facility. A *data cube* is a multidimensional array of data, where each dimension is a set of sets representing domain content, such as time or geography. The dimensions are scaled categorically, for example, region of country, state, quarter of year, and week of quarter. The cells of the cube contain aggregated measures (usually counts) of variables. To explore the data cube, one can *drill down*, *drill up*, and *drill through*. Drill down involves splitting an aggregation into subsets, e.g., splitting region of country into states. Drill up involves consolidation, i.e., aggregating subsets along a dimension. Drill through involves subsets crossing multiple sets, e.g., the user might investigate statistics within a state subset by time. Other databases and tools include object-oriented databases, transactional databases, time series and spatial databases, online analytical processing (OLAP), multidimensional OLAP (MOLAP), and relational OLAP using extended SQL (ROLAP). See Chap. 2 of *Han and Kamber [7]* for more details.

38.2.2 Data Preparation

The purpose of this step in the KDD process is to identify data quality problems, sources of noise, data redundancy, missing data, and outliers. Data quality problems can involve inconsistency with external datasets, uneven quality (e.g., if a respondent fakes an answer), and biased opportunistically collected data. Possible sources of noise include faulty data collection instruments (e.g., sensors), transmission errors (e.g., intermittent errors from satellite or Internet transmissions), data entry errors, technology limitations errors, mis-used naming conventions (e.g., using the same names for different meanings), and incorrect classification.

Redundant data exists when the same variables have different names in different databases, when a raw variable in one database is a derived variable in another, and when changes in a variable over time are not reflected in the database. These irrelevant variables impede the speed of the KDD process because dimension reduction is needed to eliminate them. Missing data may be irrelevant if we can extract useful knowledge without imputing the missing data. In addition, most statistical methods for handling missing data may fail for massive datasets, so new or modified methods still need to be developed. In detecting outliers, sophisticated methods like the Fisher information matrix or convex hull peeling are available, but are too complex for massive datasets. Although outliers may be easy to visualize

in low dimensions, high-dimensional outliers may not show up in low-dimensional projections. Currently, clustering and other statistical modeling are used.

The data preparation process involves three steps: data cleaning, database sampling, and database reduction and transformation. Data cleaning includes removal of duplicate variables, imputation of missing values, identification and correction of data inconsistencies, identification and updating of stale data, and creating a unique record (case) identification (ID). Via database sampling, the KDD process selects appropriate parts of the databases to be examined. For this to work, the data must satisfy certain conditions (e.g., no systematic biases). The sampling process can be expensive if the data have been stored in a database system such that it is difficult to sample the data the way you want and many operations need to be executed to obtain the targeted data. One must balance a trade-off between the costs of the sampling process and the mining process. Finally, database reduction is used for data cube aggregation, dimension reduction, elimination of irrelevant and redundant attributes, data compression, and encoding mechanisms via quantizations, wavelet transformation, principle components, etc.

38.3 Data Mining (DM) Models and Algorithms

The DM process is illustrated in Fig. 38.5. In this process, one will start by choosing an appropriate class of models. To fit the best model, one needs to split the sample data

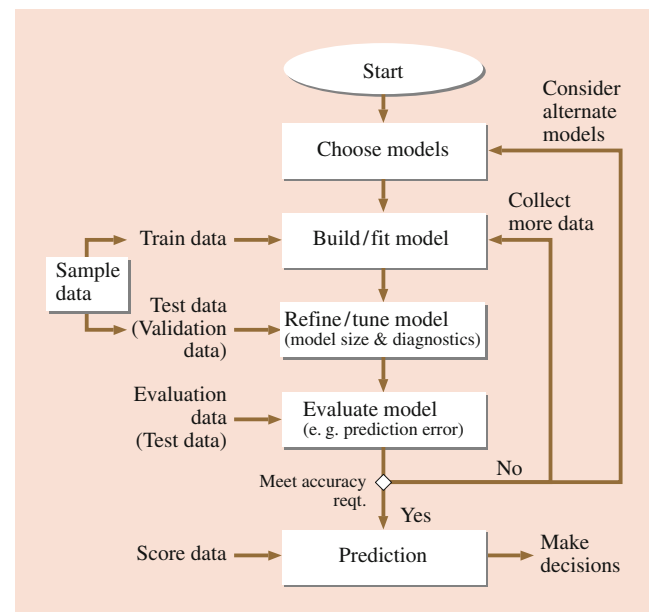


Fig. 38.5 Data mining process

into two parts: the training data and the testing data. The training data will be used to fit the model and the testing data is used to refine and tune the fitted model. After the final model is obtained, it is recommended to use an independent dataset to evaluate the goodness of the final model, such as comparing the prediction error to the accuracy requirement. (If independent data are not available, one can use the cross-validation method to compute prediction error.) If the accuracy requirement is not satisfied, then one must revisit earlier steps to reconsider other classes of models or collect additional data.

Before implementing any sophisticated DM methods, data description and visualization are used for initial exploration. Tools include descriptive statistical measures for central tendency/location, dispersion/spread, and distributional shape and symmetry; class characterizations and comparisons using analytical approaches, attribute relevance analysis, and class discrimination and comparisons; and data visualization using scatter-plot matrices, density plots, 3D stereoscopic scatter plots, and parallel coordinate plots. Following this initial step, DM methods take two forms: supervised versus unsupervised learning. Supervised learning is described as *learning with a teacher*, where the *teacher* provides data with correct answers. For example, if we want to classify online shoppers as buyers or nonbuyers using an available set of variables, our data would include actual instances of buyers and nonbuyers for training a DM method. Unsupervised learning is described as *learning without a teacher*. In this case, correct answers are not available, and DM methods would search for patterns or clusters of similarity that could later be linked to some explanation.

38.3.1 Supervised Learning

In supervised learning, we have a set of *input* variables (also known as predictors, independent variables, \mathbf{x}) that are measured or preset, and a set of *output* variables (also known as responses, dependent variables, \mathbf{y}) that are measured and assumed to be influenced by the inputs. If the outputs are continuous/quantitative, then we have a *regression* or *prediction* problem. If the outputs are categorical/qualitative, then we have a *classification* problem. First, a DM model/system is established based on the collected input and output data. Then, the established model is used to predict output values at new input values. The predicted values are denoted by $\hat{\mathbf{y}}$.

The DM perspective of *learning with a teacher* follows these steps:

- Student presents an answer (\hat{y}_i given \mathbf{x}_i)
- Teacher provides the correct answer y_i or an error e_i for the student's answer
- The result is characterized by some *loss function* or *lack-of-fit criterion*: LOF (y, \hat{y})
- The objective is to minimize the expected loss

Supervised learning includes the common engineering task of function approximation, in which we assume that the output is related to the input via some function $f(\mathbf{x}, \epsilon)$, where ϵ represents a random error, and seek to approximate $f(\cdot)$.

Below, we describe several supervised learning methods. All can be applied to both the regression and classification cases, except for those presented below under “Other Classification Methods.” We maintain the following notation. The j -th input variable is denoted by x_j (or random variable X_j) and the corresponding boldface \mathbf{x} (or \mathbf{X}) denotes the vector of p input variables $(x_1, x_2, \dots, x_p)^T$, where boldface x_i denotes the i -th sample point; N is the number of sample points, which corresponds to the number of observations of the response variable; the response variable is denoted by y (or random variable Y), where y_i denotes the i -th response observation. For the regression case, the response y is quantitative, while for the classification case, the response values are indices for C classes ($c = 1, \dots, C$). An excellent reference for these methods is *Hastie et al.* [8].

Linear and Additive Methods

In the regression case, the basic linear method is simply the *multiple linear* regression model form

$$\mu(\mathbf{x}; \beta) = E[Y | \mathbf{X} = \mathbf{x}] = \beta_0 + \sum_{m=1}^M \beta_m b_m(\mathbf{x}),$$

where the model terms $b_m(\mathbf{x})$ are prespecified functions of the input variables, for example, a simple linear term $b_m(\mathbf{x}) = x_j$ or a more complex interaction term $b_m(\mathbf{x}) = x_j x_k^2$. The key is that the model is *linear in the parameters* β . Textbooks that cover linear regression are abundant (e.g., [9, 10]). In particular, *Neter et al.* [11] provides a good background on residual analysis, model diagnostics, and model selection using best subsets and stepwise methods. In model selection, insignificant model terms are eliminated, thus the final model may be a subset of the original prespecified model. An alternate approach is to use a shrinkage method that employs a penalty function to *shrink* estimated model parameters toward zero, essentially reducing the influence of less important terms. Two options are ridge regression [12], which uses the penalty form $\sum \beta_m^2$, and the lasso [13], which uses the penalty form $\sum |\beta_m|$.

In the classification case, linear methods generate *linear decision boundaries* to separate the C classes. Although a direct linear regression approach could be applied, it is known not to work well. A better method is *logistic regression* [14], which uses log-odds (or logit transformations) of the posterior probabilities $\mu_c(\mathbf{x}) = P(Y = c | \mathbf{X} = \mathbf{x})$ for classes $c = 1, \dots, C - 1$ in the form

$$\begin{aligned}\log \frac{\mu_c(\mathbf{x})}{\mu_C(\mathbf{x})} &= \log \frac{P(Y = c | \mathbf{X} = \mathbf{x})}{P(Y = C | \mathbf{X} = \mathbf{x})} \\ &= \beta_{c0} + \sum_{j=1}^p \beta_{cj} x_j,\end{aligned}$$

$$\begin{aligned}\log \frac{\mu_c(\mathbf{x})}{\mu_C(\mathbf{x})} &= \log \frac{P(Y = c | \mathbf{X} = \mathbf{x})}{P(Y = C | \mathbf{X} = \mathbf{x})} \\ &= \beta_0 + \sum_{j=1}^p f_j(x_j),\end{aligned}$$

where the C posterior probabilities $\mu_c(\mathbf{x})$ must sum to one. The decision boundary between class $c < C$ and class C is defined by the hyperplane $\{\mathbf{x} | \beta_{c0} + \sum \beta_{cj} x_j = 0\}$, where the log-odds are zero. Similarly, the decision boundary between classes $c \neq C$ and $d \neq C$, derived from the log-odds for classes c and d , is defined by $\{\mathbf{x} | \beta_{c0} + \sum \beta_{cj} x_j = \beta_{d0} + \sum \beta_{dj} x_j\}$. In the binary case ($C = 2$), if we define $\mu(\mathbf{x}) = P(Y = 1 | \mathbf{X} = \mathbf{x})$, then $1 - \mu(\mathbf{x}) = P(Y = 2 | \mathbf{X} = \mathbf{x})$. The logit transformation is then defined as $g(\mu) = \mu/(1 - \mu)$.

Closely related to logistic regression is *linear discriminant analysis* [15], which utilizes exactly the same linear form for the log-odds ratio, and defines linear *discriminant functions* $\delta_c(\mathbf{x})$, such that \mathbf{x} is classified to class c if its maximum discriminant is $\delta_c(\mathbf{x})$. The difference between the two methods is how the parameters are estimated. Logistic regression maximizes the conditional likelihood involving the posterior probabilities $P(Y = c | \mathbf{X})$ while linear discriminant analysis maximizes the full log-likelihood involving the unconditional probabilities $P(Y = c, \mathbf{X})$. More general forms of discriminant analysis are discussed below under ‘‘Other Classification Methods.’’

Finally, it should be noted that the logistic regression model is one form of *generalized linear model* (GLM) [16]. GLM forms convert what appear to be nonlinear models into linear models, using tools such as transformations (e.g., logit) or conditioning on nonlinear parameters. This then enables the modeler to use traditional linear modeling analysis techniques. However, real data often do not satisfy the restrictive conditions of these models.

Rather than using prespecified model terms, as in a linear model, a *generalized additive model* (GAM) [17] provides a more flexible statistical method to enable modeling of nonlinear patterns in each input dimension. In the regression case, the basic GAM form is

$$\mu(\mathbf{x}) = \beta_0 + \sum_{j=1}^p f_j(x_j),$$

where the $f_j(\cdot)$ are unspecified (smooth) univariate functions, one for each input variable. The additive restriction prohibits inclusion of any interaction terms. Each function is fitted using a *nonparametric regression* modeling method, such as running-line smoothers (e.g., *lowess*, [18]), smoothing splines, or kernel smoothers [19–21]. In the classification case, an additive logistic regression model utilizes the logit transformation for classes $c = 1, \dots, C - 1$ as above

where an additive model is used in place of the linear model. However, even with the flexibility of nonparametric regression, GAM may still be too restrictive. The following sections describe methods that have essentially no assumptions on the underlying model form.

Trees and Related Methods

One DM decision tree model is *chi-square automatic interaction detection* (CHAID) [22, 23], which builds nonbinary trees using a chi-square test for the classification case and an F -test for the regression case. The CHAID algorithm first creates categorical input variables out of any continuous inputs by dividing them into several categories with approximately the same number of observations. Next, input variable categories that are not statistically different are combined, while a Bonferroni p -value is calculated for those that are statistically different. The best split is determined by the smallest p -value. CHAID continues to select splits until the smallest p -value is greater than a prespecified significance level (α).

The popular *classification and regression trees* (CART) [24] utilize recursive partitioning (binary splits), which evolved from the work of *Morgan and Sonquist* [25] and *Fielding* [26] on analyzing survey data. CARTs have a forward stepwise procedure that adds model terms and backward procedure for pruning. The model terms partition the \mathbf{x} -space into disjoint hyper-rectangular regions via indicator functions: $b^+(x; t) = 1\{x > t\}$, $b^-(x; t) = 1\{x \leq t\}$, where the *split-point* t defines the borders between regions. The resulting model terms are:

$$f_m(\mathbf{x}) = \prod_{l=1}^{L_m} b^{s_{l,m}}(x_{v(l,m)}; t_{l,m}), \quad (38.1)$$

where, L_m is the number of univariate indicator functions multiplied in the m -th model term, $x_{v(l,m)}$ is the input variable corresponding to the l -th indicator function in the m -th model term, $t_{l,m}$ is the split-point corresponding to $x_{v(l,m)}$, and $s_{l,m}$ is $+1$ or -1 to indicate the direction of the partition. The CART model form is then

$$f(\mathbf{x}; \beta) = \beta_0 + \sum_{m=1}^M \beta_m f_m(\mathbf{x}). \quad (38.2)$$

The partitioning of the \mathbf{x} -space does not keep the parent model terms because they are redundant. For example, suppose the current set has the model term:

$$f_a(\mathbf{x}) = 1\{x_3 > 7\} \cdot 1\{x_4 \leq 10\},$$

and the forward stepwise algorithm chooses to add

$$\begin{aligned} f_b(\mathbf{x}) &= f_a(\mathbf{x}) \cdot 1\{x_5 > 13\} \\ &= 1\{x_3 > 7\} \cdot 1\{x_4 \leq 10\} \cdot 1\{x_5 > 13\}. \end{aligned}$$

Then the model term $f_a(\mathbf{x})$ is dropped from the current set. Thus, the recursive partitioning algorithm follows a binary tree with the current set of model terms $f_m(\mathbf{x})$ consisting of the M leaves of the tree, each of which corresponds to a different region R_m .

In the regression case, CART minimizes the squared error loss function,

$$\text{LOF}(\hat{f}) = \sum_{i=1}^N [y_i - \hat{f}(\mathbf{x}_i)]^2,$$

and the approximation is a piecewise constant function. In the classification case, each region R_m is classified into one of the C classes. Specifically, define the proportion of class c observations in region R_m as

$$\hat{\delta}_{mc} = \frac{1}{N_m} \sum_{\mathbf{x}_i \in R_m} 1\{y_i = c\},$$

where N_m is the number of observations in the region R_m . Then the observations in region R_m are classified into the class c corresponding to the maximum proportion $\hat{\delta}_{mc}$. The algorithm is exactly the same as for regression, but with a different loss function. Appropriate choices include minimizing the misclassification error (i.e., the number of misclassified observations), the Gini index, $\sum_{c=1}^C \hat{\delta}_{mc} (1 - \hat{\delta}_{mc})$, or the deviance $\sum_{c=1}^C \hat{\delta}_{mc} \log(\hat{\delta}_{mc})$.

The exhaustive search algorithms for CART simultaneously conduct variable selection (x) and split-point selection (t). To reduce computational effort, the *fast algorithm for classification trees* [27] separates the two tasks. At each existing model term (leaf of the tree), F -statistics are calculated for variable selection. Then linear discriminant analysis is used to identify the split-point. A version for logistic and Poisson regression was presented by Chaudhuri et al. [28].

The primary drawback of CART and FACT is a bias toward selecting higher-order interaction terms due to the property of keeping only the leaves of the tree. As a consequence, these tree methods do not provide robust approximations and can have poor prediction accuracy. Loh and Shih [29] address this issue for FACT with a variant of their classification algorithm called QUEST that clusters classes into superclasses before applying linear discriminant analysis. For CART, Friedman et al. [30] introduced to the statistics literature the concepts of *boosting* [31] and *bagging* [32]

from the machine learning literature. The bagging approach generates many bootstrap samples, fits a tree to each, and then uses their average prediction. In the framework of boosting, a model term, called a *base learner*, is a *small tree* with only L disjoint regions (L is selected by the user), call it $B(\mathbf{x}, \mathbf{a})$, where \mathbf{a} is the vector of tree coefficients. The boosting algorithm begins by fitting a small tree $B(\mathbf{x}, \mathbf{a})$ to the data, and the first approximation, $\hat{f}_1(\mathbf{x})$, is then this first small tree. In the m -th iteration, residuals are calculated, then a small tree $B(\mathbf{x}, \mathbf{a})$ is fitted to the residuals and combined with the latest approximation to create the m -th approximation:

$$\begin{aligned} \hat{f}_m(\mathbf{x}; \beta_0, \beta_1, \dots, \beta_m) &= \hat{f}_{m-1}(\mathbf{x}; \beta_0, \beta_1, \\ &\dots, \beta_{m-1}) + \beta_m B(\mathbf{x}, \mathbf{a}), \end{aligned}$$

where a line search is used to solve for β_m . The resulting boosted tree, called a *multiple additive regression tree* (MART) [33], then consists of much lower-order interaction terms. Friedman [34] presents *stochastic gradient boosting*, with a variety of loss functions, in which a bootstrap-like bagging procedure is included in the boosting algorithm.

Finally, for the regression case only, *multivariate adaptive regression splines* (MARS) [35] evolved from CART as an alternative to its piecewise constant approximation. Like CART, MARS utilizes a forward stepwise algorithm to select model terms followed by a backward procedure to prune the model. A univariate version (appropriate for additive relationships) was presented by Friedman and Silverman [36]. The MARS approximation bends to model curvature at *knot* locations, and one of the objectives of the forward stepwise algorithm is to select appropriate knots. An important difference from CART is that MARS maintains the parent model terms, which are no longer redundant but are simply lower-order terms.

MARS model terms have the same form as (38.1), except the indicator functions are replaced with truncated linear functions,

$$[b^+(x; t) = [+(x - t)]_+, b^-(x; t) = [- (x - t)]_+,$$

where $[q]_+ = \max(0, q)$ and t is a univariate knot. The search for new model terms can be restricted to interactions of a maximum order (e.g., $L_m \leq 2$ permits up through two-factor interactions). The resulting MARS approximation, following (38.2), is a continuous, piecewise linear function. After selection of the model terms is completed, smoothness to achieve a certain degree of continuity may be applied.

Hastie et al. [8] demonstrate significant improvements in accuracy using MART over CART. For the regression case, comparisons between MART and MARS yield comparable results [34]. Thus, the primary decision between these two

methods is whether a piecewise constant approximation is satisfactory or if a continuous, smooth approximation would be preferred.

Artificial Neural Networks and Convolutional Neural Networks

Artificial neural network (ANN) models have been very popular for modeling a variety of physical relationships (for a general introduction see *Lippmann* [37] or *Haykin* [38]; for statistical perspectives see *White* [39], *Baron et al.* [40], *Ripley* [23], or *Cheng and Titterton* [41]). The original motivation for ANNs comes from how *learning* strengthens connections along neurons in the brain. Commonly, an ANN model is represented by a diagram of nodes in various layers with weighted connections between nodes in different layers (Fig. 38.6). At the input layer, the nodes are the input variables and at the output layer, the nodes are the response variable(s). In between, there is usually at least one *hidden* layer which induces flexibility into the modeling. *Activation functions* define transformations between layers (e.g., input to hidden). Connections between nodes can *feed back* to previous layers, but for supervised learning the typical ANN is *feedforward* only with at least one hidden layer.

The general form of a feedforward ANN with one hidden layer and activation functions $b_1(\cdot)$ (input to hidden) and $b_2(\cdot)$ (hidden to output) is

$$f_c(\mathbf{x}; \mathbf{w}, \mathbf{v}, \boldsymbol{\theta}, \gamma_c) = b_2 \left[\sum_{h=1}^H w_{hc} \cdot b_1 \left(\sum_{j=1}^p v_{jh} x_j + \theta_h \right) + \gamma_c \right], \quad (38.3)$$

where $c = 1, \dots, C$ and C is the number of output variables, p is the number of input variables, H is the number of hidden nodes, the weights v_{jh} link input nodes j to hidden

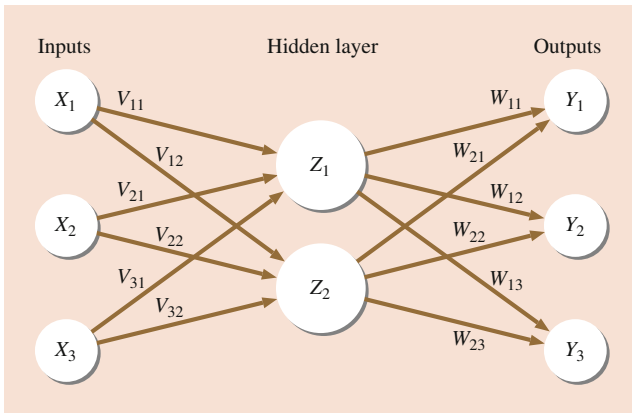


Fig. 38.6 Diagram of a typical artificial neural network for function approximation. The input nodes correspond to the input variables, and the output node(s) correspond to the output variable(s). The number of hidden nodes in the hidden layer must be specified by the user

nodes h and w_{hc} link hidden nodes h to output nodes c , and θ_h and γ_c are constant terms called bias nodes (like intercept terms). The number of coefficients to be estimated is $(p + 1)H + (H + 1)C$, which is often larger than N . The simplest activation function is a linear function $b(z) = z$, which reduces the ANN model in (38.3) with one response variable to a multiple linear regression equation. For more flexibility, the recommended activation functions between the input and hidden layer(s) are the S-shaped *sigmoidal* functions or the bell-shaped *radial basis functions*. Commonly used sigmoidal functions are the logistic function

$$b(z) = \frac{1}{1 + e^{-z}}$$

and the hyperbolic tangent

$$b(z) = \tanh(z) = \frac{1 - e^{-2z}}{1 + e^{-2z}}.$$

The most common radial basis function is the Gaussian probability density function.

In the regression case, each node in the output layer represents a quantitative response variable. The output activation function may be either a linear, sigmoidal, or radial basis function. Using a logistic activation function from input to hidden and from hidden to output, the ANN model in (38.3) becomes

$$f_c(\mathbf{x}; \mathbf{w}, \mathbf{v}, \boldsymbol{\theta}, \gamma_c) = \left[1 + \exp \left(- \sum_{h=1}^H w_{hc} z_h + \gamma_c \right) \right]^{-1},$$

where for each hidden node h

$$z_h = \left[1 + \exp \left(- \sum_{j=1}^p v_{jh} x_j + \theta_h \right) \right]^{-1}.$$

In the classification case with C classes, each class is represented by a different node in the output layer. The recommended output activation function is the *softmax* function. For output node c , this is defined as

$$b(z_1, \dots, z_C; c) = \frac{e^{z_c}}{\sum_{d=1}^C e^{z_d}}.$$

This produces output values between zero and one that sum to one and, consequently, permits the output values to be interpreted as posterior probabilities for a categorical response variable.

Mathematically, an ANN model is a nonlinear statistical model, and a nonlinear method must be used to estimate the

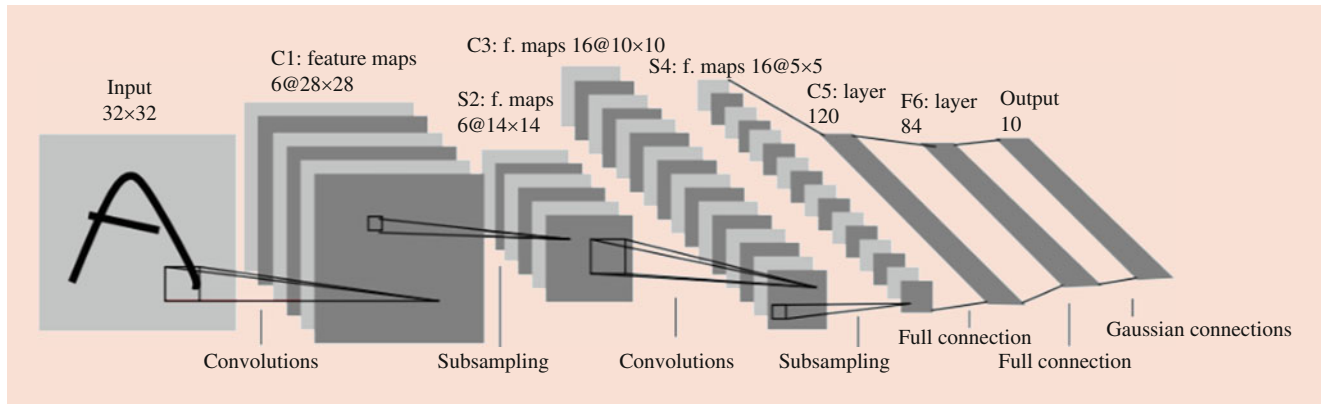


Fig. 38.7 Diagram of a typical convolutional neural network

coefficients (weights v_{jh} and w_{hc} , biases θ_h and γ_c) of the model. This estimation process is called network training. Typically, the objective is to minimize the squared error lack-of-fit criterion

$$\text{LOF}(\hat{f}) = \sum_{c=1}^C \sum_{i=1}^N [y_i - \hat{f}_c(\mathbf{x}_i)]^2.$$

The most common method for training is *backpropagation*, which is based on gradient descent. At each iteration, each coefficient (say w) is adjusted according to its contribution to the lack-of-fit

$$\Delta w = \alpha \frac{\partial (\text{LOF})}{\partial w},$$

where the user-specified α controls the step size; see *Rumelhart et al.* [42] for more details. More efficient training procedures are a subject of current ANN research.

Another major issue is the network *architecture*, defined by the number of hidden nodes. If too many hidden nodes are permitted, the ANN model will overfit the data. Many model discrimination methods have been tested, but the most reliable is validation of the model on a testing dataset separate from the training dataset. Several ANN architectures are fitted to the training dataset and then prediction error is measured on the testing dataset. Although ANNs are generally flexible enough to model anything, they are computationally intensive, and a significant quantity of representative data is required to both fit and validate the model. From a statistical perspective, the primary drawback is the overly large set of coefficients, none of which provide any intuitive understanding for the underlying model structure. In addition, since the nonlinear model form is not motivated by the true model structure, too few training data points can result in ANN approximations with extraneous nonlinearity. However, given enough good data, ANNs can outperform other modeling methods.

Convolutional neural network (CNN) [43] is a feedforward ANN network effective for pattern recognition and feature extraction of image data. The CNN combines three architectural ideas: local receptive fields, shared weights, and spatial subsampling. As in Fig. 38.7, a typical CNN for character recognition usually consists of an input layer, a convolutional layer, a pooling layer, a fully connected layer, and an output layer. Before feeding into the input plane, images of characters are size normalized and centered. In the convolutional layer, each neuron receives inputs from a set of neurons located in a small neighborhood in the previous layer. By using the same convolutional filter, the neurons share the filter weights and the number of free parameters in the convolutional layer is greatly reduced. With local receptive fields, neurons can extract elementary visual features such as corners, end points, and oriented edges. These basic features are then combined in higher layers to form useful information. Each convolutional layer is followed by an additional pooling layer which performs a local averaging and a subsampling, reducing the resolution of the feature map, as well as the sensitivity of the outputs to shifts and distortions. The CNN extracts the topological and spatial features hidden inside the image data through layer-by-layer convolution and pooling operations. These features are finally fed into the fully connected layer for classification or regression.

Support Vector Machines

Referring to the linear methods for classification described earlier, the decision boundary between two classes is a hyperplane of the form $\{\mathbf{x} | \beta_0 + \sum \beta_j x_j = 0\}$. The *support vectors* are the points that are most critical to determining the optimal decision boundary because they lie close to the points belonging to the other class. With *support vector machines* (SVM) [44], the linear decision boundary is generalized to the more flexible form

$$f(\mathbf{x}; \boldsymbol{\beta}) = \beta_0 + \sum_{m=1}^M \beta_m g_m(\mathbf{x}), \quad (38.4)$$

where the $g_m(\mathbf{x})$ are transformations of the input vector. The decision boundary is then defined by $\{\mathbf{x} | f(\mathbf{x}; \boldsymbol{\beta}) = 0\}$. To solve for the optimal decision boundary, it turns out that we do not need to specify the transformations $g_m(\mathbf{x})$, but instead require only the *kernel function* [21, 45]:

$$K(\mathbf{x}, \mathbf{x}') = \langle [g_1(\mathbf{x}), \dots, g_M(\mathbf{x})], [g_1(\mathbf{x}'), \dots, g_M(\mathbf{x}')] \rangle.$$

Two popular kernel functions for SVM are polynomials of degree d , $K(\mathbf{x}, \mathbf{x}') = (1 + \langle \mathbf{x}, \mathbf{x}' \rangle)^d$, and radial basis functions, $K(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|^2/c)$.

Given $K(\mathbf{x}, \mathbf{x}')$, we maximize the following Lagrangian dual-objective function:

$$\begin{aligned} \max_{\alpha_1, \dots, \alpha_N} \quad & \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{i'=1}^N \alpha_i \alpha_{i'} y_i y_{i'} K(\mathbf{x}_i, \mathbf{x}_{i'}) \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq \gamma, \text{ for } i = 1, \dots, N \text{ and} \\ & \sum_{i=1}^N \alpha_i y_i = 0, \end{aligned}$$

where γ is an SVM tuning parameter. The optimal solution allows us to rewrite $f(\mathbf{x}; \boldsymbol{\beta})$ as

$$f(\mathbf{x}; \boldsymbol{\beta}) = \beta_0 + \sum_{i=1}^N \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i),$$

where β_0 and $\alpha_1, \dots, \alpha_N$ are determined by solving $f(\mathbf{x}; \boldsymbol{\beta}) = 0$. The support vectors are those \mathbf{x}_i corresponding to nonzero α_i . A smaller SVM tuning parameter γ leads to more support vectors and a smoother decision boundary. A testing dataset may be used to determine the best value for γ .

The SVM extension to more than two classes solves multiple two-class problems. SVM for regression utilizes the model form in (38.4) and requires specification of a loss function appropriate for a quantitative response [8, 46]. Two possibilities are the ϵ -insensitive function

$$V_\epsilon(e) = \begin{cases} 0 & \text{if } |e| < \epsilon, \\ |e| - \epsilon & \text{otherwise,} \end{cases}$$

which ignores errors smaller than ϵ , and the *Huber* [47] function

$$V_H(e) = \begin{cases} e^2/2 & \text{if } |e| \leq 1.345, \\ 1.345|e| - e^2/2 & \text{otherwise,} \end{cases}$$

which is used in robust regression to reduce model sensitivity to outliers.

Other Classification Methods

In this section, we briefly discuss some other concepts that are applicable to DM classification problems. The basic

intuition behind a good classification method is derived from the *Bayes classifier*, which utilizes the posterior distribution $P(Y = c | \mathbf{X} = \mathbf{x})$. Specifically, if $P(Y = c | \mathbf{X} = \mathbf{x})$ is the maximum over $c = 1, \dots, C$, then \mathbf{x} would be classified to class c .

Nearest neighbor (NN) [48] classifiers seek to estimate the Bayes classifier directly without specification of any model form. The k -NN classifier identifies the k closest points to \mathbf{x} (using Euclidean distance) as the *neighborhood* about \mathbf{x} , then estimates $P(Y = c | \mathbf{X} = \mathbf{x})$ with the fraction of these k points that are of class c . As k increases, the decision boundaries become smoother; however, the *neighborhood* becomes less local (and less relevant) to \mathbf{x} . This problem of local representation is even worse in high dimensions, and modifications to the distance measure are needed to create a practical k -NN method for DM. For this purpose, *Hastie* and *Tibshirani* [49] proposed the discriminant adaptive NN distance measure to reshape the neighborhood adaptively at a given \mathbf{x} to capture the critical points to distinguish between the classes.

As mentioned earlier, linear discriminant analysis may be too restrictive in practice. *Flexible discriminant analysis* replaces the linear decision boundaries with more flexible regression models, such as GAM or MARS. *Mixture discriminant analysis* relaxes the assumption that classes are more or less spherical in shape by allowing a class to be represented by multiple (spherical) clusters; see *Hastie* et al. [50] and *Ripley* [23] for more details.

K-means clustering classification applies the K -means clustering algorithm separately to the data for each of the C classes. Each class c will then be represented by K clusters of points. Consequently, nonspherical classes may be modeled. For a new input vector \mathbf{x} , determine the closest cluster, then assign \mathbf{x} to the class associated with that cluster.

Genetic algorithms [51, 52] use processes such as genetic combination, mutation, and natural selection in an optimization based on the concepts of natural evolution. One generation of models competes to pass on characteristics to the next generation of models, until the best model is found. Genetic algorithms are useful in guiding DM algorithms, such as neural networks and decision trees [53].

38.3.2 Unsupervised Learning

Unsupervised learning, often called “learning without a teacher,” has been widely used for exploratory analysis to identify hidden patterns or groups in data. Unlike supervised learning, it draws inferences from data *without* predefined label information (i.e., response variables are not available). Given a set of observations of a random variable X , the goal of unsupervised learning is to directly infer properties of the probability density $P(X)$ without the label information.

It may be noted that labeled samples are sometimes significantly more expensive to collect (e.g., by asking human experts to make judgments) than unlabeled samples. Therefore, unsupervised learning has gained increasing interest in a variety of applications when labels are difficult to obtain, including biology, medicine, social science, business marketing, etc. In this subsection, we introduce two commonly used unsupervised learning techniques: association rules and cluster analysis.

Association Rules

Association rule analysis [8] seeks to discover co-occurrence between items in a collection and expresses such relationships as *association rules*. It is most often applied as *market basket analysis*, which deals with sales transactions to link specific products for the analysis of purchasing behaviors of customers. For example, the following rule:

$$\{\text{Diapers}\} \rightarrow \{\text{Beer}\}$$

suggests co-occurrence exists between the sale of diapers and beer. In other words, customers tend to purchase diapers and beer together. Such information is helpful for retailers to increase profit by optimizing their cross-promotion strategies, catalog design, stocking shelves, and customer relationship management.

In association analysis, a collection of one or more items is termed an *itemset*. If an itemset contains k items, it is called a k -itemset. For example, {Diapers, Beer, Eggs} is a 3-itemset. An important property of an itemset is its *support count* σ , which is defined as the number of transactions that contain this itemset. In the dataset below, the support count for {Diapers, Beer, Eggs} is two because only transactions #3 and #4 contain these three items.

TID	Items
#1	{Bread, Diapers}
#2	{Bread, Diapers, Milk, Eggs}
#3	{Diapers, Beer, Eggs}
#4	{Milk, Eggs, Beer, Diapers}
#5	{Bread, Milk, Diapers, Beer}

An *association rule* is expressed as $A \rightarrow B$, where A is called the *antecedent* and B is called the *consequent*. Given an itemset, association rule can be generated by assigning one or more items as the *antecedent* and one or more items as the *consequent*. Notably, A and B are disjoint sets, i.e., $A \cap B = \emptyset$. The importance of an association rule can be measured in terms of its *support* and *confidence*. *Support* (s) determines how frequently a rule appears among all transactions and it is defined as the support count of the rule over the total number of transactions N :

$$s(A \rightarrow B) = \frac{\sigma(A \cup B)}{N}$$

For example, support of rule {Diapers} \rightarrow {Beer} is 3/5 and support of rule {Diapers, Eggs} \rightarrow {Beer} is 2/5. Notably, a low support indicates the rule is uninteresting and customers seldom buy these items together. Minimum support (*minsup*) can be defined to eliminate uninteresting rules with $s < \text{minsup}$.

Confidence (c), on the other hand, represents how frequently items in B appear in transactions that contain A . It is defined as:

$$c(A \rightarrow B) = \frac{\sigma(A \cup B)}{\sigma(A)}$$

For example, the confidence of rule {Diapers} \rightarrow {Beer} is 3/5, and the confidence of rule {Diapers, Eggs} \rightarrow {Beer} is 2/3. A high confidence suggests it is likely for B to be present in transactions that contain A . In other words, customers tend to buy these items together and it is profitable to promote them together.

Furthermore, *lift* can be calculated as the ratio of the confidence over the expected confidence,

$$L(A \rightarrow B) = \frac{c(A \rightarrow B)}{\sigma(B)/N}$$

which, if greater than one, can be interpreted as the increased prevalence of B when associated with A . For example, if $\sigma(B)/N = 5\%$, then B is estimated to occur unconditionally 5% of the time. If $c(A \rightarrow B) = 40\%$, then given A occurs, B is estimated to occur 40% of the time. This results in a lift of 8, implying that B is 8 times more likely to occur if A occurs.

Cluster Analysis

Cluster analysis, or clustering, seeks to segment a set of objects into clusters, such that objects within a cluster are more similar to each other than those assigned to different clusters [54]. It is an unsupervised learning method since no predefined label information is needed. The goal of cluster analysis is to discover the underlying structure of the data to obtain insight into the data distribution, arrange objects into a natural hierarchy, or serve as a preprocessing step for other algorithms. Notably, defining the similarity measure among objects is critical for cluster analysis. Each object i can be represented as a set of measurements, x_{ip} ($p = 1, 2, \dots, P$), where P is the number of variables (also called attributes). Then, the pairwise distance between objects i and j is defined as $d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{p=1}^P d_p(x_{ip}, x_{jp})$, and $d_p(x_{ip}, x_{jp})$ is the dissimilarity between the values of the p th variable. For quantitative variables, the common choice for $d_p(x_{ip}, x_{jp})$ is the squared distance, i.e., $d_p(x_{ip}, x_{jp}) = (x_{ip} - x_{jp})^2$. For nonquantitative variables (e.g., ordinal or categorical variables), numerical

coding approaches [55] or Hamming distance [56] can be used. Below, we introduce some widely used algorithms for cluster analysis.

K-means [57] is one of the most popular clustering tools. It aims to partition the N objects into K clusters $C = \{C_1, C_2, \dots, C_K\}$, so that the within-cluster distance is minimized as follows:

$$\operatorname{argmin}_C \sum_{k=1}^K \sum_{i \in C_k} \|x_i - \mu_k\|^2$$

where μ_k is the mean of objects in cluster C_k . The objective function is minimized using an iterative refinement approach: In the *assignment step*, the distance between each object to the means of K clusters are calculated and the object is assigned to the cluster with minimum distance. In the *update step*, the new means of the objects in new clusters are calculated. The initialization of cluster means can be randomly selected K objects. The *assignment step* and *update step* are repeated until assignments are no longer changed. Notably, *K-means* needs to specify the desired number of clusters K , which can be selected by using prior knowledge or by trying different values and looking for the one with the most interpretable solution. Also, it is sensitive to the initialization and may converge to a local optimum. Thus, it is crucial to run the algorithm many times from multiple random starting points. The best solution is selected with the smallest value of the objective function. In addition, it is difficult for *K-means* to handle noisy data and outliers. Many algorithms, such as *K-Medoids*, were developed to improve the robustness of *K-means* [58].

Density-based clustering (DBSCAN) [59] is rooted in the idea that clusters are dense regions with objects packed together, separated by regions of lower density. It is associated with two key parameters: the radius of a neighborhood with respect to an object (ε) and the minimum number of objects to form a dense region (*MinPts*). The ε -neighborhood of object w is defined as $N_\varepsilon(w) : \{v \mid d(w, v) \leq \varepsilon\}$. Then, objects can be segmented into three groups: if $N_\varepsilon(w)$ contains at least *MinPts* objects, then the object w is a *core object*; if object w has fewer than *MinPts* objects in its ε -neighborhood, but it is in the ε -neighborhood of a core object, it is called a *border object*; and if the object is neither a core object nor a border object, it is categorized as a *noise object*. An object v is called *directly density-reachable* from an object w w.r.t. ε and *MinPts* if w is a core object and $v \in N_\varepsilon(w)$. An object v is called *density-reachable* from w w.r.t. ε and *MinPts* if there is a chain of objects o_1, o_2, \dots, o_n with $o_1 = w$ and $o_n = v$, and o_{i+1} is directly reachable from o_i . Two objects v and w are called density-connected if there is an object o such that both v and w are density-reachable from o . Then, a cluster satisfies two properties: (i) all objects within the

cluster are mutually density-connected and (ii) if an object is density-reachable from any object of the cluster, it should be included into the cluster. To find a cluster, DBSCAN starts with an object o that has not been visited. If o is a core object, then it collects all objects that are density-reachable from o and forms a cluster. Otherwise, the object is considered as noise. As opposed to *K-means*, DBSCAN is less sensitive to outliers and better handles data with arbitrary geometric shapes. However, it cannot handle data with varying density and is sensitive to parameters settings.

Hierarchical clustering [8] seeks to build a hierarchy of clusters based on pairwise dissimilarities among objects. Strategies for hierarchical clustering generally fall into two categories: agglomerative (bottom up) and divisive (top down). Agglomerative strategies start at the bottom with each object in its own cluster. It recursively merges pairs of clusters with smallest inter-cluster dissimilarity as one moves up the hierarchy. Divisive strategies start at the top with all objects in one cluster. It moves down the hierarchy by recursively splitting an existing cluster into two new clusters with the largest inter-cluster dissimilarity. Let W and V represent two clusters of objects. The dissimilarity of W and V is computed from the pairwise dissimilarities d_{wv} , where one member of the pair w is from W and the other one v is from V . *Average linkage*: $d_{AL}(W, V) = \frac{1}{N_W N_V} \sum_{v \in V} \sum_{w \in W} d_{wv}$, *complete linkage*: $d_{CL}(W, V) = \max_{v \in V, w \in W} d_{wv}$, and *single linkage*: $d_{SL}(W, V) = \min_{v \in V, w \in W} d_{wv}$ are commonly used to measure the inter-cluster dissimilarity. It is up to the user to decide when (i.e., at which level) to stop to obtain a “natural” result: objects within each cluster are sufficiently more similar to each other than to objects assigned to different clusters. The resulting hierarchical structure can be graphically represented as a *dendrogram*.

Affinity propagation (AP) [60] is based on neighbor information propagation. It finds the optimal set of class representative objects (i.e., exemplars), which make the sum of the similarities of all objects to their nearest exemplars as large as possible. For objects i, j , and k , a similarity matrix s is defined such that $s(i, j) > s(i, k)$, if object i is more similar to object j than to k . A commonly used function is $s(i, j) = -\|x_i - x_j\|^2$. The AP algorithm proceeds by letting all objects send messages to all other objects to determine exemplars with two matrices: (1) a “responsibility matrix” with element $r(i, j)$ that quantifies how well-suited object j is to be an exemplar to object i and (2) an “availability matrix” with element $a(i, j)$ that represents how appropriate it would be for object i to choose object j as its exemplar. As such, responsibility messages are sent around as: $r(i, k) \leftarrow s(i, k) - \max_{k_1 \neq k} \{a(i, k_1) + s(i, k_1)\}$. The availability messages are updated as: $a(i, k) \leftarrow$

$\min \left\{ 0, r(k, k) + \sum_{i_1 \neq i \& i_1 \neq k} \max \{0, r(i_1, k)\} \right\}$ for $i \neq k$ and $a(k, k) \leftarrow \sum_{i_1 \neq k} \max \{0, r(i_1, k)\}$. The algorithm iterates until cluster assignments are not changed. The final exemplars are chosen as those with $r(i, i) + a(i, i) > 0$. Although exemplars are similar to “centroids” generated by the *K-means*, the AP algorithm does not require the number of clusters to be predefined.

A *self-organizing map (SOM)* [61] is a type of unsupervised ANN model that represents high-dimensional input data in a low-dimensional map, preserves the topological relationship of the original data, and organizes the data according to inherent structures. Given an input $\mathbf{x} = (x_1, x_2, \dots, x_d)$, the distance to each neuron in the SOM is calculated as $d_i = \|\mathbf{w}_i - \mathbf{x}\|$, where $\mathbf{w}_i = (w_{i1}, w_{i2}, \dots, w_{id})$ is the weight of the i th neuron in the SOM (in total M neurons). The resulting distance vector $\mathbf{d} = (d_1, d_2, \dots, d_M)$ is obtained to determine the best matching neuron (BMN) by finding the smallest d_i . Then, d_i is assigned as 1 and all other elements in \mathbf{d} are set as 0. Further, the weights of the BMN and its neighbors are updated toward the input according to the rule of the Kohonen update as: $\mathbf{w}_i^{t+1} \leftarrow \mathbf{w}_i^t + h^t \cdot (\mathbf{x}^t - \mathbf{w}_i^t)$, where t is the iteration index and h^t is a neighborhood function to characterize the closeness of the BMN to other neurons in the map. In this way, a SOM arranges high-dimensional input in a two-dimensional map such that similar inputs are mapped onto neighboring regions. Thus, similar patterns of the input data are preserved. Outputs of a SOM can be characterized using a U-matrix or a Hits Map to visualize the clustering results. Conventional SOMs are designed for unsupervised learning, whereas supervised SOMs are also investigated in the literature that integrate label information as an additional element in the input vector \mathbf{x} during the training phase [62].

38.3.3 Software

Several DM software packages and libraries are available:

- SAS Enterprise Miner (www.sas.com/technologies/analytics/datamining/miner/)
- SPSS Clementine (www.spss.com/clementine/)
- XLMiner in Excel (www.xlminer.net)
- Ghostminer (www.fqspl.com.pl/ghostminer/)
- Quadstone (www.quadstone.com/)
- Insightful Miner (www.splus.com/products/iminer/)
- Statsmodels (www.statsmodels.org)
- SciKit-Learn (www.scikit-learn.org)
- Keras (www.keras.io)
- Tensorflow (www.tensorflow.org)
- Deep Learning Toolbox (www.mathworks.com/solutions/deep-learning/)
- Darknet (www.pjreddie.com/darknet/)

Haughton et al. [63] present a review of the first five listed above. The SAS and SPSS packages have the most complete set of KDD/DM tools (data handling, DM modeling, and graphics), while Quadstone is the most limited. Insightful Miner was developed by S+ [www.splus.com], but does not require knowledge of the S+ language, which is only recommended for users that are familiar with statistical modeling. For statisticians, the advantage is that Insightful Miner can be integrated with more sophisticated DM methods available with S+, such as flexible and mixture discriminant analysis. All six packages include trees and clustering, and all except Quadstone include ANN modeling. The SAS, SPSS, and XLMiner packages include discriminant analysis and association rules. Ghostminer is the only one that offers SVM tools.

SAS, SPSS, and Quadstone are the most expensive (over \$40 000) while XLMiner is a good deal for the price (under \$2 000). The disadvantage of XLMiner is that it cannot handle very large datasets. Each package has certain specializations, and potential users must carefully investigate these choices to find the package that best fits their KDD/DM needs. Below we describe some other software options for the DM modeling methods presented.

GLM or linear models are the simplest of DM tools and most statistical software can fit them, such as SAS, SPSS, S+, and Statistica [www.statsoftinc.com/]. However, it should be noted that Quadstone only offers a regression tool via scorecards, which is not the same as statistical linear models. GAM requires access to more sophisticated statistical software, such as S+.

Software for CART, MART, and MARS is available from Salford Systems [www.salford-systems.com]. SAS Enterprise Miner includes CHAID, CART, and the machine learning program C4.5 [www.rulequest.com], which uses classifiers to generate decision trees and if-then rules. SPSS Clementine and Insightful Miner also include CART, but Ghostminer and XLMiner utilize different variants of decision trees. QUEST [www.stat.wisc.edu/loh/quest.html] is available in SPSS's AnswerTree software and Statistica.

Python provides many open-source libraries for machine learning and deep learning. Statsmodels is a library that enables its users to conduct data exploration via the use of various methods of estimation of statistical models and performing statistical assertions and analysis. Scikits are additional packages of SciPy Stack designed for specific functionalities like image processing and machine learning facilitation. Keras and Tensorflow are two most prominent and convenient open-source libraries for deep learning. Keras builds neural networks at a high level of the interface. Tensorflow is developed by Google and sharpened for machine learning. It was designed to meet the high-demand requirements of Google environment for training neural networks and is a successor of DistBelief, a machine learning system based on neural networks.

Other softwares for ANN include Matlab's [<http://www.mathworks.com>] Neural Network Toolbox, Matlab's Deep Learning Toolbox, and Darknet [<http://www.pjreddie.com/darknet/>]. The Neural Network Toolbox provides a complete package for ANN modeling. The Deep Learning Toolbox supports CNN networks for classification and regression on image data. The Darknet is an open-source deep learning framework written in C and CUDA, which is fast, easy to install, and provides state-of-art methods for real-time object detection.

38.4 DM Research and Applications

Many industrial and business applications require modeling and monitoring processes with real-time data of different types: real values, categorical, and even text and image. DM is an effective tool for extracting process knowledge and discovering data patterns to provide a control aid for these processes. Advanced DM research involves complex system modeling of heterogeneous objects, where adaptive algorithms are necessary to capture dynamic system behavior. Various data mining algorithms [63], such as logistic regression, support vector machines, convolutional neural networks, decision trees, and combinations of these, have been widely adopted in practical applications. Some application examples include activity monitoring, manufacturing process modeling, object detection, health assessment, fault diagnosis, and remaining useful life prediction. DM algorithms serve as solutions to these tasks.

38.4.1 Activity Monitoring

One important DM application is the development of an effective data modeling and monitoring system for understanding customer profiles and detecting fraudulent behavior. This is generally referred to as *activity monitoring for interesting events requiring action* [65]. Other activity monitoring examples include credit card or insurance fraud detection, computer intrusion detection, some forms of fault detection, network performance monitoring, and news story monitoring.

Although activity monitoring has only recently received attention in the information industries, solutions to similar problems were developed long ago in the manufacturing industries, under the moniker *statistical process control* (SPC). SPC techniques have been used routinely for online process control and monitoring to achieve process stability and to improve process capability through variation reduction. In general, all processes are subject to some natural variability regardless of their state. This natural variability is usually

small and unavoidable and is referred to as *common cause variation*. At the same time, processes may be subject to other variability caused by improper machine adjustment, operator errors, or low-quality raw material. This variability is usually large, but avoidable, and is referred to as *special cause variation*. The basic objective of SPC is to detect the occurrence of special cause variation (or process shifts) quickly, so that the process can be investigated and corrective action may be taken before quality deteriorates and defective units are produced. The main ideas and methods of SPC were developed in the 1920s by Walter Shewhart of Bell Telephone Laboratories and have had tremendous success in manufacturing applications [66]. *Montgomery* and *Woodall* [67] provide a comprehensive panel discussion on SPC, and multivariate methods are reviewed by *Hayter* and *Tsui* [68] and *Mason et al.* [69].

Although the principle of SPC can be applied to service industries, such as business process monitoring, fewer applications exist for two basic reasons that *Montgomery* identified. First, the system that needs to be monitored and improved is obvious in manufacturing applications, while it is often difficult to define and observe in service industries. Second, even if the system can be clearly specified, most nonmanufacturing operations do not have natural measurement systems that reflect the performance of the system. However, these obstacles no longer exist, due to the many natural and advanced measurement systems that have been developed. In the telecommunications industry, for example, advanced software and hardware technologies make it possible to record and process huge amounts of daily data in business transactions and service activities. These databases contain potentially useful information to the company that may not be discovered without knowledge extraction or DM tools.

While SPC ideas can be applied to business data, SPC methods are not directly applicable. Existing SPC theories are based on small- or medium-sized samples, and the basic hypothesis testing approach is intended to detect only simple shifts in a process mean or variance. Recently, *Jiang et al.* [70] successfully generalized the SPC framework to model and track thousands of diversified customer behaviors in the telecommunication industry. The challenge is to develop an integrated strategy to monitor the performance of an entire multistage system and to develop effective and efficient techniques for detecting the systematic changes that require action.

A dynamic business process can be described by the dynamic linear models introduced by *West* [71],

$$\text{Observation equation: } X_t = A_t \theta_t + \Delta_t,$$

$$\text{System evolution equation: } \theta_t = B_t \theta_{t-1} + \Lambda_t,$$

$$\text{Initial information: } \pi(S_0),$$

where A_t and B_t represent observation and state transition matrices, respectively, and Δ_t and Λ_t represent observation and system transition errors, respectively. Based on the dynamic system model, a model-based process monitoring and root-cause identification method can be developed. Monitoring and diagnosis include fault pattern generation and feature extraction, isolation of the critical processes, and root-cause identification. Jiang et al. [70] utilize this for individual customer prediction and monitoring. In general, individual modeling is computationally intractable and cluster models should be developed with mixture distributions [72].

One particularly competitive industry is telecommunications. Since divestiture and government deregulation, various telephone services, such as cellular, local and long distance, domestic, and commercial, have become battle grounds for telecommunication service providers. Because of the data and information-oriented nature of the industry, DM methods for knowledge extraction are critical. To remain competitive, it is important for companies to develop business planning systems that help managers make good decisions. In particular, these systems will allow sales and marketing people to establish successful customer loyalty programs for churn prevention and to develop fraud detection modules for reducing revenue loss through market segmentation and customer profiling.

A major task in this research is to develop and implement DM tools within the business planning system. The objectives are to provide guidance for targeting business growth, to forecast year-end usage volume and revenue growth, and to value risks associated with the business plan periodically. Telecommunication business services include voice and non-voice services, which can be further categorized to include domestic, local, international, products, toll-free calls, and calling cards. For usage forecasting, a minutes growth model is utilized to forecast domestic voice usage. For revenue forecasting, the average revenue per minute on a log scale is used as a performance measure and is forecasted by a double exponential smoothing growth function. A structural model is designed to decompose the business growth process into three major subprocesses: add, disconnect, and base. To improve explanatory power, the revenue unit is further divided into different customer groups. To compute confidence and prediction intervals, bootstrapping and simulation methods are used.

To understand the day effect and seasonal effect, the concept of bill-month equivalent business days (EBD) is defined and estimated. To estimate EBD, the factor characteristics of holidays (non-EBD) are identified and eliminated and the day effect is estimated. For seasonality, the US Bureau of the Census X-11 seasonal adjustment procedure is used.

38.4.2 Mahalanobis-Taguchi System

Genichi Taguchi is best known for his work on robust design and design of experiments. The Taguchi robust design methods have generated a considerable amount of discussion and controversy and are widely used in manufacturing [73–76]. The general consensus among statisticians seems to be that, while many of Taguchi's overall ideas on experimental design are very important and influential, the techniques he proposed are not necessarily the most effective statistical methods. Nevertheless, Taguchi has made significant contributions in the area of quality control and quality engineering. For DM, Taguchi has recently popularized the *Mahalanobis-Taguchi System* (MTS), a new set of tools for diagnosis, classification, and variable selection. The method is based on a Mahalanobis distance scale that is utilized to measure the level of abnormality in *abnormal* items as compared to a group of *normal* items. First, it must be demonstrated that a Mahalanobis distance measure based on all available variables is able to separate the abnormal from the normal items. Should this be successfully achieved, orthogonal arrays and signal-to-noise ratios are used to select an *optimal* combination of variables for calculating the Mahalanobis distances.

The MTS method has been claimed to be very powerful for solving a wide range of problems, including manufacturing inspection and sensing, medical diagnosis, face and voice recognition, weather forecasting, credit scoring, fire detection, earthquake forecasting, and university admissions. Two recent books have been published on the MTS method by Taguchi et al. [77] and Taguchi and Jugulum [78]. Many successful case studies in MTS have been reported in engineering and science applications in many large companies, such as Nissan Motor Co., Mitsubishi Space Software Co., Xerox, Delphi Automotive Systems, ITT Industries, Ford Motor Company, Fuji Photo Film Company, and others. While the method is getting a lot of attention in many industries, very little research [79] has been conducted to investigate how and when the method is appropriate.

38.4.3 Manufacturing Process Modeling

One area of DM research in manufacturing industries is quality and productivity improvement through DM and knowledge discovery. Manufacturing systems nowadays are often very complicated and involve many manufacturing process stages where hundreds or thousands of in-process measurements are taken to indicate or initiate process control of the system. For example, a modern semiconductor manufacturing process typically consists of over 300 steps, and in each

step multiple pieces of equipment are used to process the wafer. Inappropriate understanding of interactions among in-process variables will create inefficiencies at all phases of manufacturing, leading to long product/process realization cycle times and long development times, resulting in excessive system costs.

Current approaches to DM in electronics manufacturing include neural networks, decision trees, Bayesian models, and rough set theory [80, 81]. Each of these approaches carries certain advantages and disadvantages. Decision trees, for instance, produce intelligible rules and hence are very appropriate for generating process control or design of experiments strategies. They are, however, generally prone to outlier and imperfect data influences. Neural networks, on the other hand, are robust against data abnormalities but do not produce readily intelligible knowledge. These methods also differ in their ability to handle high-dimensional data, to discover arbitrarily shaped clusters [57] and to provide a basis for intuitive visualization [82]. They can also be sensitive to training and model building parameters [59]. Finally, the existing approaches do not take into consideration the localization of process parameters. The patterns or clusters identified by existing approaches may include parameters from a diverse set of components in the system. Therefore, a combination of methods that complement each other to provide a complete set of desirable features is necessary.

It is crucial to understand process structure and yield components in manufacturing, so that problem localization can permit reduced production costs. For example, semiconductor manufacturing practice shows that over 70% of all fatal defects and close to 90% of yield excursions are caused by problems related to process equipment [83]. Systematic defects can be attributed to many categories that are generally associated with technologies and combinations of different process operations. To implement DM methods successfully for knowledge discovery, some future research for manufacturing process control must include yield modeling, defect modeling, and variation propagation.

Yield Modeling

In electronics manufacturing, the ANSI standards [84] and practice generally assume that the number of defects on an electronics product follows a Poisson distribution with mean λ . The Poisson random variable is an approximation of the sum of independent Bernoulli trials, but defects on different components may be correlated since process yield critically depends on product groups, process steps, and types of defects [85]. Unlike traditional defect models, an appropriate logit model can be developed as follows. Let the number of defects of category X on an electronics product be

$$U_X = \sum Y_X$$

and

$$\begin{aligned} \text{logit}[E(Y_X)] &= \alpha_X^0 + \alpha_X^O \cdot O_X \\ &+ \alpha_X^C \cdot C_X + \alpha_X^{OC} \cdot O_X \cdot C_X, \end{aligned}$$

where $\text{logit}(z) = \log[z/(1 - z)]$ is the link function for Bernoulli distributions, and Y_X is a Bernoulli random variable representing a defect from defect category X . The default logit of the failure probability is α_X^0 , and α_X^O and α_X^C are the main effects of operations (O_X) and components (C_X). Since the Y_X s are correlated, this model will provide more detailed information about defects.

Multivariate Defect Modeling

Since different types of defects may be caused by the same operations, multivariate Poisson models are necessary to account for correlations among different types of defects. The *trivariate reduction method* suggests an additive Poisson model for the vector of Poisson counts $U = (U_1, U_2, \dots, U_k)'$,

$$U = AV,$$

where A is a matrix of zeros and ones, and $V = (v_1, v_2, \dots, v_p)'$ consists of independent Poisson variables v_i . The variance-covariance matrix takes the form $\text{Var}(U) = A \Sigma A' = \Phi + \nu \nu'$, where $\Phi = \text{diag}(\mu_i)$ is a diagonal matrix with the mean of the individual series, and ν is the common covariance term. Note that the v_i are essentially latent variables, and a factor analysis model can be developed for analyzing multivariate discrete Poisson variables such that

$$\log[E(U)] = \mu + L \cdot F,$$

where U is the vector of defects, L is the matrix of factor loadings, and F contains common factors representing effects of specific operations. By using factor analysis, it is possible to relate product defects to the associated packages and operations.

Multistage Variation Propagation

Inspection tests in an assembly line usually have functional overlap, and defects from successive inspection stations exhibit strong correlations. Modeling serially correlated defect counts is an important task for defect localization and yield prediction. Poisson regression models, such as the generalized event-count method [86] and its alternatives, can be utilized to account for serial correlations of defects in different inspection stations. Factor analysis methods based on hidden Markov models [87] can also be constructed to investigate how variations are propagated through assembly lines.

38.4.4 Object Detection

One area of DM image research is object detection, a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class in digital images and videos. An object-class detection is to localize and extract information of all objects in an image that belongs to a given class. Well-studied domains of object detection include face detection and pedestrian detection.

Face detection, as a specific case of object-class detection, focuses on the detection of frontal human faces. It is analogous to image detection in which the image of a person is matched bit by bit. Image matches with the image stores in database. Any facial feature changes in the database will invalidate the matching process. The main difficulties in face detection includes severe occlusion and variation of head poses.

Pedestrian detection provides fundamental information for semantic understanding of the video footages and considered an essential and significant in intelligent video surveillance system. It has an obvious extension to automotive applications due to enhance road safety and is offered as an advanced driver assistant system option by many car manufacturers in 2017. The major challenges of pedestrian detection arise from different possible posture, various appearance styles, the presence of occluding accessories, and frequent occlusions among pedestrians.

Existing methods for object detection generally fall into machine learning approaches and deep learning approaches. For machine learning approaches, it first extracts handcrafted features such as edges, corners, colors, etc. of the region of interest cropped by sliding window, then uses classifiers such as SVM to do the classification. On the other hand, deep learning approaches use CNN to extract the image features and thus able to do end-to-end object detection without specifically defining features. Ross et al. [88] prompted *regions with CNN* (R-CNN) to achieve dramatic improvements in accuracy of objects detection, which can be seen as a major breakthrough in the field of object detection. Subsequently, a series of R-CNN-based detection methods such as Fast R-CNN [89] and Faster R-CNN [90] were proposed. Those are classic two-stage object detection approaches that usually include a region proposal localization stage and a network classification stage. Then single shot detection methods, such as *single shot detector* (SSD) [91], RetinaNet [92], *you only look once* (YOLO) [93–95], etc., are developed. They skip the region proposal stage and run detection directly over a dense sampling of possible locations through a single CNN. The “one-stage” detection methods treat the detection as a single regression problem and are faster and simpler.

38.4.5 Surveillance of Public Health Systems

Public health surveillance is another important DM application. The objective of public health surveillance is to examine health trends, detect changes in disease incidence and death rates, and to plan, implement, and evaluate public health practice by systematically collecting, analyzing, and interpreting public health data (chronic or infectious diseases). Understanding challenges to nations’ public health systems and how those challenges shift over time is of crucial importance for policymakers to establish effective strategies. In public health surveillance, the volume and velocity of data streams have dramatically grown in recent decades. In spite of the growing data volume, advances in information technology have enabled collection of cause-of-death data in a more timely manner.

The availability of public health big data provides a comprehensive picture of health system status in terms of the causes of significant population-wide changes, the underlying risks, the changes in the pattern of health-related losses, etc. Numerous efforts have been made to monitor and evaluate the health of populations by taking advantage of public health big data and data mining techniques. For example, Google flu trend (GFT) is a data analytics model developed by Google for predicting weekly reported influenza-like illness (ILI) rates using instant query data [96]. However, as reported in Refs. [96, 97], GFT failed to provide accurate predictions, and predicted more than double the actual rate of doctor visits for ILI reported by the Centers for Disease Control and Prevention during the 2012–2013 season. The model’s failure has led to a large number of research papers aiming to improve its predictive accuracy [98–100]. One representative method was ARGO, proposed in Ref. [98], which not only incorporated seasonality in historical ILI rates, but also captured changes in the public’s online searching behaviors over time.

38.5 Concluding Remarks

While DM and KDD methods are gaining recognition and have become very popular in many companies and enterprises, the success of these methods is still somewhat limited. Below, we discuss a few obstacles.

First, the success of DM depends on a close collaboration of subject-matter experts and data modelers. In practice, it is often easy to identify the right subject-matter expert, but difficult to find the qualified data modeler. While the data modeler must be knowledgeable and familiar with DM methods, it is more important to be able to formulate real problems such that the existing methods can be applied. In reality, traditional academic training mainly focuses on knowledge of modeling

algorithms and lacks training in problem formulation and interpretation of results. Consequently, many modelers are very efficient in fitting models and algorithms to data, but have a hard time determining when and why they should use certain algorithms. Similarly, the existing commercial DM software systems include many sophisticated algorithms, but there is a lack of guidance on which algorithms to use.

Second, implementation of DM is difficult to apply effectively across an industry. Although it is clear that extracting hidden knowledge and trends across an industry would be useful and beneficial to all companies in the industry, it is typically impossible to integrate the detailed data from competing companies due to confidentiality and proprietary issues. Currently, the industry practice is that each company will integrate their own detailed data with the more general, aggregated industry-wide data for knowledge extraction. It is obvious that this approach will be significantly less effective than the approach of integrating the detailed data from all competing companies. It is expected that, if these obstacles can be overcome, the impact of the DM and KDD methods will be much more prominent in industrial and commercial applications.

References

- Berry, M.J.A., Linoff, G.: *Mastering Data Mining: The Art and Science of Customer Relationship Management*. Wiley, New York (2000)
- Wegman, E.: *Data Mining Tutorial, Short Course Notes, Interface 2001 Symposium*. Cosa Mesa, Californien (2001)
- Adriaans, P., Zantinge, D.: *Data Mining*. Addison-Wesley, New York (1996)
- Friedman, J.H.: *Data Mining and Statistics: What Is the Connection?* Technical Report. Stat. Dep., Stanford University (1997)
- Clark, K.B., Fujimoto, T.: Product development and competitiveness. *J. Jpn. Int. Econ.* **6**(2), 101–143 (1992)
- LaBahn, D.W., Ali, A., Krapfel, R.: New product development cycle time. The influence of project and process factors in small manufacturing companies. *J. Bus. Res.* **36**(2), 179–188 (1996)
- Han, J., Kamber, M.: *Data Mining: Concept and Techniques*. Morgan Kaufmann, San Francisco (2001)
- Hastie, T., Friedman, J.H., Tibshirani, R.: *Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, Berlin/Heidelberg/New York (2001)
- Weisberg, S.: *Applied Linear Regression*. Wiley, New York (1980)
- Seber, G.: *Multivariate Observations*. Wiley, New York (1984)
- Neter, J., Kutner, M.H., Nachtsheim, C.J., Wasserman, W.: *Applied Linear Statistical Models*, 4th edn. Irwin, Chicago (1996)
- Hoerl, A.E., Kennard, R.: Ridge regression: biased estimation of nonorthogonal problems. *Technometrics*. **12**, 55–67 (1970)
- Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B.* **58**, 267–288 (1996)
- Agresti, A.: *An Introduction to Categorical Data Analysis*. Wiley, New York (1996)
- Hand, D.: *Discrimination and Classification*. Wiley, Chichester (1981)
- McCullagh, P., Nelder, J.A.: *Generalized Linear Models*, 2nd edn. Chapman Hall, New York (1989)
- Hastie, T., Tibshirani, R.: *Generalized Additive Models*. Chapman Hall, New York (1990)
- Cleveland, W.S.: Robust locally-weighted regression and smoothing scatterplots. *J. Am. Stat. Assoc.* **74**, 829–836 (1979)
- Eubank, R.L.: *Spline Smoothing and Nonparametric Regression*. Marcel Dekker, New York (1988)
- Wahba, G.: *Spline Models for Observational Data, Applied Mathematics*, vol. 59. SIAM, Philadelphia (1990)
- Härdle, W.: *Applied Non-parametric Regression*. Cambridge University Press, Cambridge (1990)
- Biggs, D., deVilleville, B., Suen, E.: A method of choosing multiway partitions for classification and decision trees. *J. Appl. Stat.* **18**(1), 49–62 (1991)
- Ripley, B.D.: *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge (1996)
- Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: *Classification and Regression Trees*. Wadsworth, Belmont (1984)
- Morgan, J.N., Sonquist, J.A.: Problems in the analysis of survey data, and a proposal. *J. Am. Stat. Assoc.* **58**, 415–434 (1963)
- Fielding, A.: Binary segmentation: the automatic interaction detector and related techniques for exploring data structure. In: O’Muircheartaigh, C.A., Payne, C. (eds.) *The Analysis of Survey Data, Volume I: Exploring Data Structures*, pp. 221–258. Wiley, New York (1977)
- Loh, W.Y., Vanichsetakul, N.: Tree-structured classification via generalized discriminant analysis. *J. Am. Stat. Assoc.* **83**, 715–728 (1988)
- Chaudhuri, W.D.L., Loh, W.Y., Yang, C.C.: Generalized Regression Trees. *Stat. Sin.* **5**, 643–666 (1995)
- Loh, W.Y., Shih, Y.S.: Split-selection methods for classification trees. *Stat. Sin.* **7**, 815–840 (1997)
- Friedman, J.H., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. *Ann. Stat.* **28**, 337–407 (2000)
- Freund, Y., Schapire, R.: Experiments with a new boosting algorithm, machine learning. In: Kaufmann, M. (ed.) *Proceedings of the Thirteenth International Conference, Bari, Italy*, pp. 148–156 (1996)
- Breiman, L.: Bagging predictors. *Mach. Learn.* **26**, 123–140 (1996)
- Friedman, J.H.: Greedy function approximation: a gradient boosting machine. *Ann. Stat.* **29**, 1189–1232 (2001)
- Friedman, J.H.: Stochastic gradient boosting. *Comput. Stat. Data Anal.* **38**(4), 367–378 (2002)
- Friedman, J.H.: Multivariate adaptive regression splines (with discussion). *Ann. Stat.* **19**, 1–141 (1991)
- Friedman, J.H., Silverman, B.W.: Flexible parsimonious smoothing and additive modeling. *Technometrics*. **31**, 3–39 (1989)
- Lippmann, R.P.: An Introduction to Computing with Neural Nets, *IEEE ASSP Magazine*, 4–22 April (1987)
- Haykin, S.S.: *Neural Networks: A Comprehensive Foundation*, 2nd edn. Prentice Hall, Upper Saddle River (1999)
- White, H.: Learning in neural networks: a statistical perspective. *Neural Comput.* **1**, 425–464 (1989)
- Barron, A.R., Barron, R.L., Wegman, E.J.: Statistical learning networks: a unifying view, computer science and statistics. In: Wegman, E.J., Gantz, D.T., Miller, J.J. (eds.) *Proceedings of the 20th Symposium on the Interface 1992*, pp. 192–203. American Statistical Association, Alexandria (1992)
- Cheng, B., Titterton, D.M.: Neural networks: a review from a statistical perspective (with discussion). *Stat. Sci.* **9**, 2–54 (1994)
- Rumelhart, D., Hinton, G., Williams, R.: Learning internal representations by error propagation. In: Rumelhart, D.E., McClelland, J.L. (eds.) *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*, vol. 1: Foundations, pp. 318–362. MIT, Cambridge (1986)

43. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE*. **86**(11), 2278–2324 (1998)
44. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. *Knowledge Discovery and Data Mining*. **2**(2), 121–167 (1998)
45. Shawe-Taylor, J., Cristianini, N.: *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge (2004)
46. Cristianini, N., Shawe-Taylor, J.: *An Introduction to Support Vector Machines*. Cambridge University Press, Cambridge (2000)
47. Huber, P.: Robust estimation of a location parameter. *Ann. Math. Stat.* **53**, 73–101 (1964)
48. Dasarathy, B.V.: *Nearest Neighbor Pattern Classification Techniques*. IEEE Comput. Soc., New York (1991)
49. Hastie, T., Tibshirani, R.: Discriminant adaptive nearest-neighbor classification. *IEEE Trans. Pattern Mach. Intell.* **18**, 607–616 (1996)
50. Hastie, T., Tibshirani, R., Buja, A.: Flexible discriminant and mixture models. In: Kay, J., Titterton, M. (eds.) *Statistics and Artificial Neural Networks*. Oxford University Press, Oxford (1998)
51. Koza, J.R.: *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT, Cambridge (1992)
52. Banzhaf, W., Nordin, P., Keller, R.E., Francone, F.D.: *Genetic Programming: An Introduction*. Morgan Kaufmann, San Francisco (1998)
53. Smith, P.W.H.: Genetic programming as a data-mining tool. In: Abbass, H.A., Sarker, R.A., Newton, C.S. (eds.) *Data Mining: a Heuristic Approach*, pp. 157–173. Idea Group Publishing, London (2002)
54. Gordon, A.: *Classification*, 2nd edn. Chapman Hall, New York (1999)
55. Ralambondrainy, H.: A conceptual version of the *K*-means algorithm. *Pattern Recogn. Lett.* **16**, 1147–1157 (1995)
56. Zhang, P., Wang, X., Song, P.: Clustering categorical data based on distance vectors. *J. Am. Stat. Assoc.* **101**, 355–367 (2006)
57. Hartigan, J.A., Wong, M.A.: A *K*-means clustering algorithm. *Appl. Stat.* **28**, 100–108 (1979)
58. Park, H., Jun, C.: A simple and fast algorithm for *K*-Medoids clustering. *Expert Syst. Appl.* **36**, 3336–3341 (2009)
59. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering cluster in large spatial databases. In: *Proceedings of 1996 International Conference on Knowledge Discovery and Data Mining (KDD96)*, Portland, 226–231 (1996)
60. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. *Science*. **315**(5814), 972–976 (2007)
61. Kohonen, T.: *Self-Organization and Associative Memory*, 3rd edn. Springer, Berlin Heidelberg New York (1989)
62. Chen, Y., Yang, H.: Self-organized neural network for the quality control of 12-lead ECG signals. *Physiol. Meas.* **33**, 1399–1418 (2012)
63. Houghton, D., Deichmann, J., Eshghi, A., Sayek, S., Teebagy, N., Topi, H.: A review of software packages for data mining. *Am. Stat.* **57**(4), 290–309 (2003)
64. Friedman, J., Hastie, T., Tibshirani, R.: *The Elements of Statistical Learning*, vol. 1: Springer Series in Statistics, New York (2001)
65. Fawcett, T., Provost, F.: Activity monitoring: noticing interesting changes in behavior. In: *Proceedings of KDD-99, San Diego 1999*, pp. 53–62, San Diego (1999)
66. Woodall, W.H., Tsui, K.-L., Tucker, G.R.: A review of statistical and fuzzy quality control based on categorical data. *Front. Stat. Qual. Control.* **5**, 83–89 (1997)
67. Montgomery, D.C., Woodall, W.H.: A discussion on statistically-based process monitoring and control. *J. Qual. Technol.* **29**, 121–162 (1997)
68. Hayter, A.J., Tsui, K.-L.: Identification and qualification in multivariate quality control problems. *J. Qual. Technol.* **26**(3), 197–208 (1994)
69. Mason, R.L., Champ, C.W., Tracy, N.D., Wierda, S.J., Young, J.C.: Assessment of multivariate process control techniques. *J. Qual. Technol.* **29**, 140–143 (1997)
70. Jiang, W., Au, S.-T., Tsui, K.-L.: A statistical process control approach for customer activity monitoring, Technical Report, AT&T Labs (2004)
71. West, M., Harrison, J.: *Bayesian Forecasting and Dynamic Models*, 2nd edn. Springer, New York (1997)
72. Fraley, C., Raftery, A.E.: Model-based clustering, discriminant analysis, and density estimation. *J. Am. Stat. Assoc.* **97**, 611–631 (2002)
73. Taguchi, G.: *Introduction to Quality Engineering: Designing Quality into Products and Processes*. Asian Productivity Organization, Tokyo (1986)
74. Nair, V.N.: Taguchi's parameter design: a panel discussion. *Technometrics*. **34**, 127–161 (1992)
75. Tsui, K.-L.: An overview of Taguchi method and newly developed statistical methods for robust design. *IIE Trans.* **24**, 44–57 (1992)
76. Tsui, K.-L.: A critical look at Taguchi's modeling approach for robust design. *J. Appl. Stat.* **23**, 81–95 (1996)
77. Taguchi, G., Chowdhury, S., Wu, Y.: *The Mahalanobis–Taguchi System*. McGraw-Hill, New York (2001)
78. Taguchi, G., Jugulum, R.: *The Mahalanobis–Taguchi Strategy: A Pattern Technology System*. Wiley, New York (2002)
79. Woodall, W.H., Koudelik, R., Tsui, K.-L., Kim, S.B., Stoumbos, Z.G., Carvounis, C.P.: A review and analysis of the Mahalanobis–Taguchi system. *Technometrics*. **45**(1), 1–15 (2003)
80. Kusiak, A., Kurasek, C.: Data mining of printed-circuit board defects. *IEEE Trans. Robot. Autom.* **17**(2), 191–196 (2001)
81. Kusiak, A.: Rough set theory: a data mining tool for semiconductor manufacturing. *IEEE Trans. Electron. Packag. Manuf.* **24**(1), 44–50 (2001)
82. Ultsch, A.: *Information and Classification: Concepts, Methods and Applications*. Springer, Berlin Heidelberg New York (1993)
83. Wong, A.Y.: A statistical approach to identify semiconductor process equipment related yield problems. In: *IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems, Paris 1997*, pp. 20–22. IEEE Computer Society, Paris (1997)
84. ANSI: Am. Nat. Standards Institute, IPC-9261, In-Process DPMO and Estimated Yield for PWB (2002)
85. Baron, M., Lakshminarayan, C.K., Chen, Z.: Markov random fields in pattern recognition for semiconductor manufacturing. *Technometrics*. **43**, 66–72 (2001)
86. King, G.: Event count models for international relations: generalizations and applications. *Int. Stud. Q.* **33**(2), 123–147 (1989)
87. Smyth, P.: Hidden Markov models for fault detection in dynamic systems. *Pattern Recogn.* **27**(1), 149–164 (1994)
88. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 580–587 (2014)
89. Girshick, R.: Fast R-CNN. In: *Proceedings of the IEEE international conference on computer vision, 1440–1448* (2015)
90. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 91–99 (2015)
91. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: single shot MultiBox detector. *Lect. Notes Comput. Sci.* 21–37 (2016)

92. Ye, T., Wang, B., Song, P., Li, J.: Automatic railway traffic object detection system using feature fusion refine neural network under shunting mode. *Sensors*. **18**, 1916 (2018)
93. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
94. Redmon, J., Farhadi, A.: YOLO9000: Better, Faster, Stronger. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
95. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
96. Lazer, D., Kennedy, R., King, G., Vespignani, A.: The parable of Google Flu: traps in big data analysis. *Science*. **343**, 1203–1205 (2014)
97. Butler, D.: When Google got flu wrong. *Nature*. **494**, 155 (2013)
98. Yang, S., Santillana, M., Kou, S.: ARGO: a model for accurate estimation of influenza epidemics using Google search data. *Proc. Natl. Acad. Sci.* (2015)
99. Copeland, P., Romano, R., Zhang, T., Hecht, G., Zigmond, D., Stefansen, C.: Google disease trends: an update. *Nature*. **457**, 1012–1014 (2013)
100. Santillana, M., Zhang, D.W., Althouse, B.M., Ayers, J.W.: What can digital disease detection learn from (an external revision to) Google Flu Trends? *Am. J. Prev. Med.* **47**, 341–347 (2014)



Kwok-Leung Tsui is a chair professor in the School of Data Science at the City University of Hong Kong. He has a Ph.D. in statistics from the University of Wisconsin at Madison. Dr. Tsui is an (elected) fellow of American Statistical Association and American Society for Quality, and was a recipient of the NSF Young Investigator Award. He is currently the departmental editor of the *IISE Transactions* and a member of the Management Committee for *Technometrics*.



Victoria Chen is a professor in the Department of Industrial, Manufacturing, and Systems Engineering and director of the Center on Stochastic Modeling, Optimization, and Statistics at the University of Texas at Arlington. She is currently serving as an executive secretary on the INFORMS Board. She is a guest coeditor for the *Annals of Operations Research*.



Wei Jiang is a distinguished professor in Antai College of Economics and Management at Shanghai Jiao Tong University. He obtained his Ph.D. degree in industrial engineering and engineering management from Hong Kong University of Science and Technology in 2000. Prior to joining Shanghai Jiao Tong University, he worked as a statistical consultant at AT&T Labs, Morristown. His current research activities include statistical methods for quality control, data mining, and enterprise intelligence.



Fangfang Yang is a research fellow in School of Data Science at the City University of Hong Kong. She earned her Ph.D. in system engineering and engineering management from the City University of Hong Kong in 2017. Current research areas of Dr. Yang include prognostics and health management, degradation modeling, and deep learning.



Chen Kan is an assistant professor in the Department of Industrial, Manufacturing, and Systems Engineering, the University of Texas at Arlington. He earned his Ph.D. in industrial and manufacturing engineering from the Pennsylvania State University in 2018. Current research areas of Dr. Kan include complex systems modeling and monitoring, smart manufacturing and IoT, and health informatics. He is a member of IEEE, IISE, and INFORMS.



Qing Mai and Xin Zhang

Contents

39.1	Background	817
39.1.1	Popularity and Challenges.....	817
39.1.2	Basics.....	818
39.1.3	The CANDECOMP/PARAFAC Decomposition.....	819
39.1.4	The Tensor Normal Distribution.....	819
39.1.5	Notation and Organization.....	820
39.2	The Tensor Generalized Linear Model	820
39.2.1	Background: The Generalized Linear Model.....	820
39.2.2	Model.....	820
39.2.3	Estimation.....	821
39.3	Envelope Tensor Response Regression	821
39.3.1	Multivariate Response Envelope Model.....	821
39.3.2	Model.....	822
39.3.3	Estimation.....	822
39.3.4	A Technical Remark.....	823
39.4	Tensor Envelope Partial Least Squares Regression	824
39.4.1	Partial Least Squares Regression.....	824
39.4.2	A Tensor PLS Algorithm.....	824
39.4.3	A Population Model.....	825
39.5	Covariate-Adjusted Tensor Classification in High Dimensions	825
39.5.1	Linear Discriminant Analysis.....	825
39.5.2	Model.....	825
39.5.3	Estimation.....	826
39.6	A Real Data Example	827
References		827

a challenging topic in multivariate statistics. In this book chapter, we aim to review the current literature on statistical models and methods for tensor data analysis.

Keywords

Discriminant analysis · Dimension reduction · Generalized linear model · Multivariate linear regression · Tensor data

39.1 Background

39.1.1 Popularity and Challenges

In many contemporary datasets, the measurements on each observation are stored in the form of a tensor. For example, in neuroimaging studies, electroencephalography (EEG) data are collected as two-way tensors (i.e., matrices), where the rows correspond to electrodes placed at different locations on the scalp, and the columns correspond to times points. The anatomical magnetic resonance image (MRI) is a three-way tensor, as it measures the activities of brains in the 3D space. By adding the time domain, functional MRI is a four-way tensor. Other research areas that frequently involve tensor data include computational biology, personalized recommendation, signal processing, computer vision, graph analysis, and data mining [42].

However, the popularity of tensor data is accompanied by many challenges in the analysis. We highlight two of them here. On the one hand, tensor data are often of intimidating dimensions. For example, in the benchmark EEG data (<http://kdd.ics.uci.edu/databases/eeg/eeg.data.html>), each subject is measured by 64 channels of electrodes at 256 time points. The resulting 64×256 matrix has a total number of 16,384

Abstract

This book chapter provides a brief introduction of tensors and a selective overview of tensor data analysis. Tensor data analysis has been an increasingly popular and also

Q. Mai · X. Zhang (✉)
Department of Statistics, Florida State University, Tallahassee, FL, USA
e-mail: mai@stat.fsu.edu; henry@stat.fsu.edu

Research is partially supported by CCF-1617691, CCF-1908969 and DMS-1613154, from the U.S. National Science Foundation.

elements in total. The modeling of such a high-dimensional dataset is generally impossible with classical statistical methods. Moreover, in many applications we have tensors of higher orders, and the issue of high dimensionality is even more severe.

On the other hand, there is a lack of analysis methods that directly target tensor data. Most familiar statistical methods are designed for vector data. While one can always force a tensor to be a vector, we lose the information contained in the tensor structure in doing so. To see this, again consider the EEG data as an example. In principle, we can stack the columns in the data matrix to form a long vector and proceed with vector-based methods. However, each column in the matrix corresponds to measurements at the same time point. Once we transform the matrix to a vector, we no longer have such an intuitive interpretation for the data. The statistical analysis is thus susceptible to loss of efficiency for the same reason. Consequently, although there have been impressive developments in high-dimensional statistics [25, 26, 28, 29, 47, 51, 55, 57, 72, 83, 88, 98, 99, e.g.], we are still in demand of tensor data analysis methods.

These challenges have motivated many research efforts on tensor data analysis. A central interest in such research is the control of model complexity by leveraging the tensor structure. Usually this is achieved by imposing low-rank, sparse, and/or separable covariance assumptions. In Sects. 39.2–39.5, we review several methods that resort to such assumptions. But we first introduce some basics for tensor that are necessary to understand the methods in this chapter. A comprehensive review on tensor operation can be found in [42].

39.1.2 Basics

For a positive integer M , an M -way tensor is an object in $\mathbb{R}^{p_1 \times \dots \times p_M}$, where \mathbb{R} denotes the field of real numbers and p_m 's are positive integers. It is easy to see that a vector is a one-way tensor, and a matrix is a two-way tensor.

For an M -way tensor $\mathbf{A} \in \mathbb{R}^{p_1 \times \dots \times p_M}$, p_m is its dimension along the m -th mode. We can vectorize \mathbf{A} to $\text{vec}(\mathbf{A}) \in \mathbb{R}^p$, where $p = \prod_{m=1}^M p_m$. The element $a_{i_1 \dots i_M}$ becomes the j -th element in $\text{vec}(\mathbf{A})$, where $j = 1 + \sum_{m=1}^M (i_m - 1) \prod_{m'=1}^{m-1} p_{m'}$. Similarly, a tensor can be unfolded along a mode to form a matrix. The mode- m matricization of \mathbf{A} is denoted as $\mathbf{A}_{(m)}$, a $p_m \times \prod_{j \neq m} p_j$ matrix. The element $a_{i_1 \dots i_M}$ in \mathbf{A} is mapped to be the (i_m, j) element in $\mathbf{A}_{(m)}$, where $j = 1 + \sum_{k=1, k \neq m}^M (i_k - 1) J_k$ with $J_k = \sum_{m' \neq m} p_{m'}$. The norm of \mathbf{A} is denoted as $\|\mathbf{A}\| = \sqrt{\sum_{j_1, \dots, j_M} a_{j_1 \dots j_M}^2} = \sqrt{\text{vec}^T(\mathbf{A}) \text{vec}(\mathbf{A})}$. For two tensors \mathbf{A}, \mathbf{B} of the same dimensions, their inner product is defined as $\langle \mathbf{A}, \mathbf{B} \rangle = \text{vec}^T(\mathbf{A}) \text{vec}(\mathbf{B})$.

A tensor can be multiplied with matrices or vectors along each mode. For $\mathbf{G} \in \mathbb{R}^{d \times p_m}$, we define the mode- m matrix product of \mathbf{A} with \mathbf{G} to be $\mathbf{D} = \mathbf{A} \times_m \mathbf{G} \in \mathbb{R}^{p_1 \times \dots \times p_{m-1} \times p_{m+1} \times \dots \times p_M}$ if and only if $\mathbf{D}_{(m)} = \mathbf{G} \mathbf{A}_{(m)}$. Similarly, we define the mode- m vector product of \mathbf{A} with a vector $\mathbf{v} \in \mathbb{R}^{p_m}$ to be $\mathbf{A} \bar{\times}_m \mathbf{v} \in \mathbb{R}^{p_1 \times \dots \times p_{m-1} \times p_{m+1} \times \dots \times p_M}$, with the $(i_1, \dots, i_{m-1}, i_{m+1}, \dots, i_M)$ element being $\sum_{i_m=1}^{p_m} a_{i_1 \dots i_M} v_{i_m}$.

In particular, for matrices $\mathbf{G}_m \in \mathbb{R}^{d_m \times p_m}$ and M -way tensor $\mathbf{C} \in \mathbb{R}^{d_1 \times \dots \times d_M}$, if $\mathbf{A} = \mathbf{C} \times_1 \mathbf{G}_1 \times_2 \mathbf{G}_2 \dots \times_M \mathbf{G}_M \in \mathbb{R}^{p_1 \times \dots \times p_M}$, we use the shorthand notation

$$\mathbf{A} = \llbracket \mathbf{C}; \mathbf{G}_1, \dots, \mathbf{G}_M \rrbracket. \quad (39.1)$$

Alternatively, (39.1) is also called the Tucker decomposition of \mathbf{A} . See Fig. 39.1 for an illustration when \mathbf{A} and \mathbf{C} are three-way tensors.

The matrix Kronecker product often appears in tensor literature. For two matrices $\mathbf{F} \in \mathbb{R}^{I \times J}$, $\mathbf{G} \in \mathbb{R}^{K \times L}$, we define their Kronecker product to be

$$\mathbf{F} \otimes \mathbf{G} = \begin{pmatrix} f_{11} \mathbf{G} & f_{12} \mathbf{G} & \dots & f_{1J} \mathbf{G} \\ \vdots & \vdots & \ddots & \vdots \\ f_{I1} \mathbf{G} & f_{I2} \mathbf{G} & \dots & f_{IJ} \mathbf{G} \end{pmatrix} \in \mathbb{R}^{(IK) \times (JL)}. \quad (39.2)$$

For a series of matrix $\mathbf{G}_m, m = 1, \dots, M$, $\mathbf{G} = \otimes_{m=1}^M \mathbf{G}_m$ is invertible if and only if \mathbf{G}_m^{-1} exists for $m = 1, \dots, M$. Moreover, $\mathbf{G}^{-1} = \otimes_{m=1}^M \mathbf{G}_m^{-1}$. With Kronecker product, we note a useful fact about the Tucker decomposition. If $\mathbf{A}, \mathbf{C}, \mathbf{G}_m, m = 1, \dots, M$ satisfy (39.1), then we must have that $\text{vec}(\mathbf{A}) = (\mathbf{G}_M \otimes \dots \otimes \mathbf{G}_1) \text{vec}(\mathbf{C}) = (\otimes_{m=1}^M \mathbf{G}_m) \text{vec}(\mathbf{C})$, where \otimes is the Kronecker product.

Another related matrix product is the Khatri-Rao product. For $\mathbf{F} \in \mathbb{R}^{I \times K}$, $\mathbf{G} \in \mathbb{R}^{J \times K}$, their Khatri-Rao product is

$$\mathbf{F} \circ \mathbf{G} = [\mathbf{f}_1 \otimes \mathbf{g}_1 \quad \dots \quad \mathbf{f}_M \otimes \mathbf{g}_M] \in \mathbb{R}^{(IJ) \times K}. \quad (39.3)$$

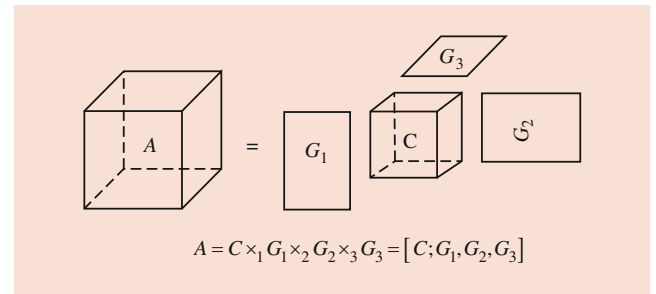


Fig. 39.1 An illustration of the Tucker decomposition $\mathbf{A} = \llbracket \mathbf{C}; \mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3 \rrbracket$

39.1.3 The CANDECOMP/PARAFAC Decomposition

In this section, we highlight the CANDECOMP/PARAFAC (CP) decomposition that serves as the foundation of many statistical methods [41]. It can be viewed as a generalization of singular value decomposition for matrices.

Consider $\mathbf{X} \in \mathbb{R}^{p_1 \times \dots \times p_M}$. For a pre-chosen positive integer R , the CP decomposition looks for R sets of vectors $\mathbf{a}_r^{(m)}$, $m = 1, \dots, M$ that jointly best approximate \mathbf{X} . In other words, for each (j_1, \dots, j_M) , we expect

$$x_{j_1 \dots j_M} \approx \sum_{r=1}^R a_{rj_1}^{(1)} a_{rj_2}^{(2)} \dots a_{rj_M}^{(M)}. \tag{39.4}$$

Two equivalent shorthand notations for the CP decomposition are

$$\mathbf{X} \approx \sum_{r=1}^R \mathbf{a}_r^{(1)} \circ \dots \circ \mathbf{a}_r^{(M)} = \llbracket \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(M)} \rrbracket, \tag{39.5}$$

where $\mathbf{A}^{(m)} = [\mathbf{a}_1^{(m)} \dots \mathbf{a}_R^{(m)}]$. See Fig. 39.2 for a demonstration of CP decomposition on a three-way tensor.

The CP decomposition is formally recast as the following minimization problem:

$$\begin{aligned} \{\mathbf{a}_r^{(m)}\}_{m=1, \dots, M; r=1, \dots, R} &= \arg \min \|\mathbf{X} - \widehat{\mathbf{X}}\|^2, \\ \text{s.t. } \widehat{\mathbf{X}} &= \llbracket \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(M)} \rrbracket. \end{aligned} \tag{39.6}$$

This problem can be solved iteratively over the $\mathbf{A}^{(m)}$'s. When $\mathbf{A}^{(m')}$ are fixed for all $m' \neq m$, (39.6) reduces to a least squares problem over $\mathbf{A}^{(m)}$. This iterative algorithm is known as the alternating least squares (ALS) [8, 37]. Since the introduction of the ALS algorithm, many other efforts have been spent on alternative ways to perform the CP decomposition [62, 73, 74], but the ALS algorithm seems to remain the most popular.

The CP decomposition is widely used on tensor data in a way much like principal component analysis on vector data. Many researchers obtain a multilinear reduction of \mathbf{X} and

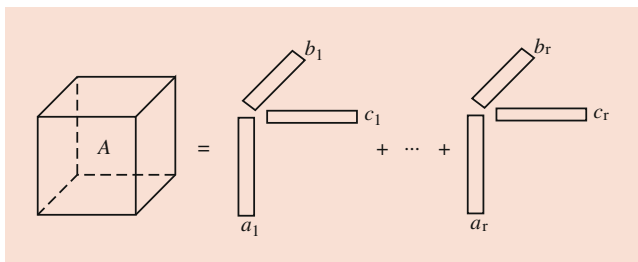


Fig. 39.2 CP decomposition of a three-way tensor \mathbf{X}

then perform analysis on the reduced predictor [9, 52, 87, e.g.]. Alternatively, one could use the CP decomposition to reduce the number of parameters, as the tensor generalized linear mode we discuss in Sect. 39.2.

39.1.4 The Tensor Normal Distribution

We will repeatedly use the tensor normal distribution that generalizes the matrix normal distribution [35]. A tensor $\mathbf{Z} \in \mathbb{R}^{p_1 \times \dots \times p_M}$ is said to be a standard tensor normal random variable if each element in \mathbf{Z} has the univariate standard normal distribution. Then, for any \mathbf{X} of the same dimension of \mathbf{Z} , we say \mathbf{X} is tensor normal, i.e., $\mathbf{X} \sim TN(\boldsymbol{\mu}; \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_M)$, if

$$\mathbf{X} = \boldsymbol{\mu} + \llbracket \mathbf{Z}; \boldsymbol{\Sigma}_1^{1/2}, \dots, \boldsymbol{\Sigma}_M^{1/2} \rrbracket, \tag{39.7}$$

where $\boldsymbol{\mu} \in \mathbb{R}^{p_1 \times \dots \times p_M}$ and $\boldsymbol{\Sigma}_m \in \mathbb{R}^{p_m \times p_m}$ are symmetric positive definite matrices (Fig. 39.3). The parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}_m$ are similar to the mean and covariance parameters for the multivariate normal distribution. In particular, $\boldsymbol{\mu} = E(\mathbf{X})$ is the expectation of \mathbf{X} , and

$$\boldsymbol{\Sigma}_m \propto E \{ (\mathbf{X} - \boldsymbol{\mu})_{(m)} (\mathbf{X} - \boldsymbol{\mu})_{(m)}^T \} \tag{39.8}$$

characterizes the dependence structure along the m -th mode of \mathbf{X} .

The tensor normal distribution is related to the multivariate normal distribution by the fact that $\text{vec}(\mathbf{X}) = \text{vec}(\boldsymbol{\mu}) + \boldsymbol{\Sigma}^{1/2} \text{vec}(\mathbf{Z})$, where $\boldsymbol{\Sigma} = \otimes_{m=1}^M \boldsymbol{\Sigma}_m \in \mathbb{R}^{p \times p}$, where $p = \prod_{m=1}^M p_m$. In other words, $\text{vec}(\mathbf{X}) \sim N(\text{vec}(\boldsymbol{\mu}), \boldsymbol{\Sigma})$. Note that, although $\boldsymbol{\Sigma}$ is of dimension $p \times p$, it is fully determined by $\sum_{m=1}^M p_m(p_m + 1)/2$ free parameters.

A minor issue with the tensor normal distribution is, though, that the covariances $\boldsymbol{\Sigma}_m$ are not identifiable themselves. We can change their scales without affecting the tensor normal distribution. For example, when $M = 2$, $TN(0; \boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2)$ is exactly the same as $TN(0; c\boldsymbol{\Sigma}_1, c^{-1}\boldsymbol{\Sigma}_2)$ for any $c > 0$. In practice, researchers often resolve this issue by fixing the scale of all the covariance matrices except for 1. For

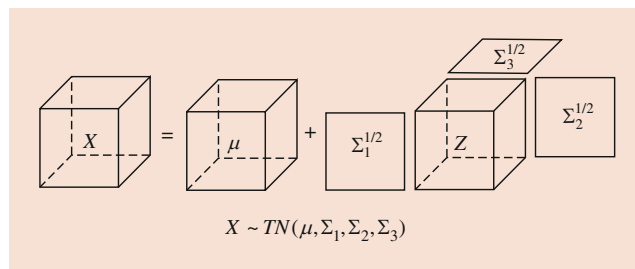


Fig. 39.3 A demonstration for tensor normal distribution when \mathbf{X} is a three-way tensor

example, we could require the Frobenius norms of Σ_m , $m < M$ to be 1, and only allow Σ_M to have an arbitrary Frobenius norm. Alternatively, we could require $\sigma_{m,11} = 1$, $m < M$. In the following sections, we do not reiterate on this issue unless necessary, as this is not a central problem in tensor modeling.

39.1.5 Notation and Organization

Throughout the rest of this chapter, a scalar is written as a lower-case letter (e.g., x), a vector is written as a lower-case letter in boldface (\mathbf{x}), and a tensor (2-way and above) is written as a capital letter in boldface (\mathbf{X}). For any positive integer r , \mathbf{I}_r denotes the identity matrix with dimension $r \times r$. The notation $\mathbf{X} \perp \mathbf{Y}$ means that the random variables \mathbf{X} and \mathbf{Y} are independent, while $\mathbf{X} \stackrel{d}{=} \mathbf{Y}$ means that \mathbf{X} and \mathbf{Y} have the same distribution. For ease of presentation, if a matrix is of the form $\Phi = \Phi_M \otimes \cdots \otimes \Phi_1$, we denote $\Phi_{-m} = \Phi_M \otimes \cdots \otimes \Phi_{m+1} \otimes \Phi_{m-1} \otimes \cdots \otimes \Phi_1$.

In Sect. 39.2 we introduce generalized linear models with tensor predictors. Section 39.3 contains an introduction to the tensor response regression; a tensor partial least squares (PLS) algorithm is presented in Sect. 39.4. In Sect. 39.5 we discuss classification based on a mixture of tensor and vector predictors. In each section we first introduce the related vector methods and then proceed to tensor data. We emphasize that the research in tensor data analysis has witnessed rapid developments in the recent years [3, 63, 69, 75, 84, 86, 87, e.g.]. It is impossible to conduct an exhaustive review, but we list some papers that propose methods closely related to those we introduce in Sects. 39.2–39.5. Finally, a real data example is given in Sect. 39.6.

Regression with tensor predictors (related to Sect. 39.2) is one of the most studied topics in tensor analysis. See [34, 40, 49, 53, 63, 64, 70, 76, 77, 89, 92, 95, 97]. Tensor response regression (related to Sect. 39.3) is studied in [44, 68, 80]. Tensor PLS algorithms (related to Sect. 39.4) are proposed in [5, 24, 93]. Classification methods (related to Sect. 39.5) are developed in [1, 43, 50, 71, 82, 85, 94].

39.2 The Tensor Generalized Linear Model

39.2.1 Background: The Generalized Linear Model

The generalized linear model (GLM) includes a family of most popular statistical models for vector data, with linear regression and logistic regression as two important examples. Consider $\{Y, \mathbf{X}\}$, where $Y \in \mathbb{R}$ is the response and $\mathbf{X} \in \mathbb{R}^p$ is the vector predictor. GLM assumes that Y is drawn from an exponential family with a probability mass or density function of the form

$$f(y | \theta, \phi) = \exp \left\{ \frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right\}, \quad (39.9)$$

where θ is the natural parameter and $\phi > 0$ is the dispersion parameter. Denote $\mu = E(Y | \mathbf{X})$ and g as a strictly increasing link function. GLM assumes that

$$g(\mu) = a + \boldsymbol{\beta}^\top \mathbf{X},$$

where $\boldsymbol{\beta} \in \mathbb{R}^p$. The parameter of interest, $\boldsymbol{\beta}$, is often estimated by the maximum likelihood estimator (MLE). For most generalized linear models such as logistic regression and Poisson regression, the MLE does not have an explicit form, but efficient algorithms have been developed to find the solution. More recently, there have been considerable interests in regularized estimators for GLM, where penalty functions are added to enhance stability and/or enforce variable selection [28, 72, 99, e.g.].

When the predictor \mathbf{X} is a tensor instead of a vector, in principle we could fit GLM on $Y, \text{vec}(\mathbf{X})$. However, this brutal-force approach leads to an overwhelmingly large number of parameters. In most applications we do not have sufficient samples to estimate all these parameters, and the computation could be painfully slow. In note of this issues, tensor GLM is proposed to control the number of parameters by utilizing the tensor structure. Also, tensor GLM allows us to model vector and tensor predictors simultaneously, as we discuss in Sect. 39.2.2.

39.2.2 Model

Consider $\{Y, \mathbf{U}, \mathbf{X}\}$, where $Y \in \mathbb{R}$ is the response, $\mathbf{U} \in \mathbb{R}^q$ is the vector predictor, and $\mathbf{X} \in \mathbb{R}^{p_1 \times \cdots \times p_M}$ is the tensor predictor. Just as in GLM, we assume that Y is drawn from an exponential family with the probability mass or density function in (39.9). Similarly, define $\mu = E(Y | \mathbf{U}, \mathbf{X})$ and a strictly increasing link function $g(\mu)$. Tensor GLM assumes that

$$g(\mu) = \beta_0 + \boldsymbol{\gamma}^\top \mathbf{U} + \left\langle \sum_{r=1}^R \boldsymbol{\beta}_1^{(r)} \circ \boldsymbol{\beta}_2^{(2)} \circ \cdots \circ \boldsymbol{\beta}_M^{(r)}, \mathbf{X} \right\rangle, \quad (39.10)$$

where $r \geq 1$ is a positive integer, $\boldsymbol{\gamma} \in \mathbb{R}^q$, and $\boldsymbol{\beta}_m^{(r)} \in \mathbb{R}^{p_m}$ for $r = 1, \dots, R$. Obviously, tensor GLM inherits the nice interpretation of GLM that the predictors have linear effects on $g(\mu)$. As a concrete example, one may consider the simplest case where $R = 1, M = 2$. Now \mathbf{X} is a matrix, and the model in (39.10) reduces to

$$g(\mu) = \beta_0 + \boldsymbol{\gamma}^\top \mathbf{U} + \boldsymbol{\beta}_1^\top \mathbf{X} \boldsymbol{\beta}_2, \quad (39.11)$$

where $\beta_1 \in \mathbb{R}^{p_1}$ is the effect across rows of \mathbf{X} , while $\beta_2 \in \mathbb{R}^{p_2}$ is the effect across the columns of \mathbf{X} . In this sense, β_1, β_2 jointly reduces \mathbf{X} by row and column. A similar idea has also been proposed in the framework of the so-called dimension folding on tensor data [45], although dimension folding is proposed for sufficient dimension reduction [6, 14–16, 46, 48, 56, 96].

To further see that tensor GLM reduces the number of parameters by honoring the tensor structure, we compare it to GLM on $\{Y, \mathbf{U}, \text{vec}(\mathbf{X})\}$. The vectorized GLM is equivalent to

$$g(\mu) = \beta_0 + \boldsymbol{\gamma}^\top \mathbf{U} + \langle \mathbf{B}, \mathbf{X} \rangle, \quad (39.12)$$

where $\mathbf{B} \in \mathbb{R}^{p_1 \times \dots \times p_M}$; recall that $\langle \mathbf{B}, \mathbf{X} \rangle = \text{vec}^\top(\mathbf{B})\text{vec}(\mathbf{X})$. It is now apparent that the tensor GLM is a special case of the vectorized GLM by requiring that the coefficient tensor \mathbf{B} has a rank- R CP decomposition, i.e., $\mathbf{B} = \sum_{r=1}^R \beta_1^{(r)} \circ \beta_2^{(r)} \circ \dots \circ \beta_M^{(r)}$. The low-rank assumption on \mathbf{B} reduces the number of free parameters from $q + \prod_{m=1}^M p_m$ to $q + R \sum_m p_m$.

39.2.3 Estimation

Similar to classical GLM, tensor GLM is estimated by maximizing the likelihood function. Let

$$\mathbf{B} = \sum_{r=1}^R \beta_1^{(r)} \circ \beta_2^{(r)} \circ \dots \circ \beta_M^{(r)} = \llbracket \mathbf{B}_1, \dots, \mathbf{B}_M \rrbracket,$$

where $\mathbf{B}_m = [\beta_m^{(1)}, \dots, \beta_m^{(R)}] \in \mathbb{R}^{p_m \times R}$. The log-likelihood function for tensor GLM is

$$l(\alpha, \boldsymbol{\gamma}, \mathbf{B}_1, \dots, \mathbf{B}_M) = \sum_{i=1}^n \frac{y_i \theta_i - b(\theta_i)}{a(\phi)} + \sum_{i=1}^n c(y_i, \phi), \quad (39.13)$$

where θ_i is linked to the parameters through the model in (39.10).

Although $g(\mu)$ is not linear in $(\mathbf{B}_1, \dots, \mathbf{B}_M)$, it is linear in \mathbf{B}_m when we fix $\mathbf{B}_{m'}$ for $m' \neq m$. More specifically, it can be shown that

$$\begin{aligned} & \left\langle \sum_{r=1}^R \beta_1^{(r)} \circ \beta_2^{(r)} \circ \dots \circ \beta_M^{(r)}, \mathbf{X} \right\rangle \\ &= \langle \mathbf{B}_m, \mathbf{X}_{(m)} (\mathbf{B}_M \odot \dots \odot \mathbf{B}_{m+1} \odot \mathbf{B}_{m-1} \odot \dots \odot \mathbf{B}_1) \rangle, \end{aligned}$$

where we recall that \odot is the Khatri-Rao product between matrices.

It follows that, given $\mathbf{B}_{m'}, m' \neq m$, we can fit a classical GLM over $\{Y, \mathbf{U}, \text{vec}(\widehat{\mathbf{X}}^{(m)})\}$ to obtain \mathbf{B}_m , where $\widehat{\mathbf{X}}^{(m)} = \mathbf{X}_{(m)} (\mathbf{B}_M \odot \dots \odot \mathbf{B}_{m+1} \odot \mathbf{B}_{m-1} \odot \dots \odot \mathbf{B}_1)$. Iteratively updating

\mathbf{B}_m yields our final estimator. Note that, in each step of this iterative algorithm, we only need to solve for the parameter \mathbf{B}_m of dimension $p_m \times R$, which is usually efficient, especially when R is small. Hence, the iterative algorithm can fit models on large tensors within a reasonably short amount of time.

If we want to further boost the stability in our estimation of tensor GLM, regularization can be applied just as in classical GLM. Denote $P_\alpha(t, \lambda)$ as a penalty function, where λ is the tuning parameter, and α is an index for the penalty function. For example, the elastic net penalty is $P_\alpha(t, \lambda) = \lambda[(\alpha - 1)t^2/2 + (2 - \lambda)t]$, $\lambda \in [1, 2]$. The regularized estimator is defined as the maximizer of

$$l(\beta_0, \boldsymbol{\gamma}, \mathbf{B}_1, \dots, \mathbf{B}_M) - \sum_{m=1}^M \sum_{r=1}^R \sum_{j=1}^{p_m} P_\alpha(|\beta_{mj}^{(r)}|, \lambda). \quad (39.14)$$

The regularized estimator can be found by an iterative algorithm as well. When we fix $\mathbf{B}_{m'}, m' \neq m$, \mathbf{B}_m is the regularized estimator for GLM over $\{Y, \mathbf{U}, \text{vec}(\widehat{\mathbf{X}}^{(m)})\}$. For various penalty functions, efficient algorithms have been developed for their combination with the classical GLM. These algorithms can be readily borrowed to implement regularized tensor GLM.

We close this section by listing some theoretical properties for tensor GLM. We focus on tensor GLM without regularization. The consistency of tensor GLM involves an identifiability condition that does not appear in classical GLM. Note that the CP decomposition of \mathbf{B} is not identifiable in general. We can permute and/or rescale the basis $\mathbf{B}_1, \dots, \mathbf{B}_M$ without changing the coefficient tensor \mathbf{B} . To resolve this issue, we consider the parameter space

$$\mathcal{B} = \left\{ (\mathbf{B}_1, \dots, \mathbf{B}_M) : \beta_{m1}^{(r)} = 1, \text{ for } m = 1, \dots, M, \right. \\ \left. r = 1, \dots, R \text{ and } \beta_{M1}^{(1)} > \dots > \beta_{M1}^{(R)} \right\},$$

which is open and convex. If we restrict our attention to this parameter space and further assume that the parameters are identifiable, it can be shown that $\text{vec}(\widehat{\mathbf{B}}_1, \dots, \widehat{\mathbf{B}}_M)$ are \sqrt{n} -consistent estimators for $\text{vec}(\mathbf{B}_1, \dots, \mathbf{B}_M)$.

39.3 Envelope Tensor Response Regression

39.3.1 Multivariate Response Envelope Model

In this section, we review the multivariate response regression that motivates the tensor response regression. We also briefly discuss the envelope assumption that will be developed in parallel for tensor data. Consider $\{Y, \mathbf{X}\}$, where $Y \in \mathbb{R}^q$ is the response and $\mathbf{X} \in \mathbb{R}^p$ is the predictor. The classical multivariate response regression model is

$$\mathbf{Y} = \boldsymbol{\beta}\mathbf{X} + \boldsymbol{\epsilon}, \tag{39.15}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{q \times p}$ is the coefficient matrix and $\boldsymbol{\epsilon} \in \mathbb{R}^q$ is the statistical error independent of \mathbf{X} with $E\boldsymbol{\epsilon} = 0$ and $\text{var}(\boldsymbol{\epsilon}) = \boldsymbol{\Sigma} \in \mathbb{R}^{q \times q}$. Without loss of generality, we assume that the intercept is zero. The parameter of interest $\boldsymbol{\beta}$ can be estimated with the ordinary least squares (OLS) estimator. If we further assume that $\boldsymbol{\epsilon}$ is normal, the OLS estimator is also the MLE, although statisticians often use the OLS estimator even without the normality assumption.

The envelope assumption is more recently proposed to improve estimation efficiency. When the dimension of \mathbf{Y} , q , is large, likely not all of them are related to the predictor \mathbf{X} . Hence, the envelope assumption indicates that the response is only related to the predictor when refined on a low-dimensional subspace. In other words, there exists a subspace $\mathcal{S} \subseteq \mathbb{R}^q$ such that

$$\mathbf{Q}_{\mathcal{S}}\mathbf{Y} \mid \mathbf{X} \stackrel{d}{=} \mathbf{Q}_{\mathcal{S}}\mathbf{Y}, \quad \mathbf{Q}_{\mathcal{S}}\mathbf{Y} \perp \mathbf{P}_{\mathcal{S}}\mathbf{Y} \mid \mathbf{X}, \tag{39.16}$$

where $\mathbf{P}_{\mathcal{S}}$ is the projection on to \mathcal{S} and $\mathbf{Q}_{\mathcal{S}} = \mathbf{I}_q - \mathbf{P}_{\mathcal{S}}$. These assumptions imply that the subspace \mathcal{S} contains all the material information for the regression, while its complement only brings in variability that degrades our inference about it is desirable to identify \mathcal{S} before estimating $\boldsymbol{\beta}$.

The multivariate response envelope model can be estimated by MLE. A key step is rewriting the envelope assumption in (39.16) into a new parameterization of $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$. With some linear algebra it can be shown that (39.16) is equivalent to that $\text{span}(\boldsymbol{\beta}) \subset \mathcal{S}$ and $\boldsymbol{\Sigma} = \mathbf{P}_{\mathcal{S}}\boldsymbol{\Sigma}\mathbf{P}_{\mathcal{S}} + \mathbf{Q}_{\mathcal{S}}\boldsymbol{\Sigma}\mathbf{Q}_{\mathcal{S}}$. Hence, we can estimate the coordinate of \mathcal{S} and estimate the other parameters accordingly. For more background on [12, 13, 17, 18, 66, 67, 90].

39.3.2 Model

On tensor data, it often makes sense to treat the tensor as the response and perform regression on vector predictors. For example, in the EEG data, we may want to compare the brain activities for subjects with and without recent alcohol consumption. Such comparison can be viewed as a regression problem with the EEG tensor as the response and an indicator function of alcohol consumption as the predictor. The coefficients in these model characterize the difference between EEGs. In some other applications, additional covariates may be available as well, such as age and gender. These covariates can be included as part of the vector predictor to eliminate possible confounding factors.

For the tensor response $\mathbf{Y} \in \mathbb{R}^{q_1 \times \dots \times q_M}$ and the vector predictor $\mathbf{X} \in \mathbb{R}^p$, we consider the tensor response linear model:

$$\mathbf{Y} = \mathbf{B} \bar{\times}_{(M+1)} \mathbf{X} + \boldsymbol{\epsilon}, \tag{39.17}$$

where $\mathbf{B} \in \mathbb{R}^{q_1 \times \dots \times q_M \times p}$ is the parameter of interest and $\boldsymbol{\epsilon} \in \mathbb{R}^{q_1 \times \dots \times q_M}$. We assume that $\boldsymbol{\epsilon} \sim TN(0; \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_M)$, although the normality is not essential. Rather, the main advantage in this assumption is that we determine the covariance of $\boldsymbol{\epsilon}$ with a smaller number of parameters in $\boldsymbol{\Sigma}_m, m = 1, \dots, M$. Such an approach is common in the tensor literature; see Sects. 39.4 and 39.5 for example. Other works that involve similar assumptions include [30, 39].

Similar to tensor GLM, estimation of \mathbf{B} is difficult without additional assumptions, because \mathbf{B} is extremely high dimensional. To this end, we further assume that the parameter \mathbf{B} has the envelope structure. Let $\mathbb{P}(\mathbf{Y}) = \llbracket \mathbf{Y}; \mathbf{P}_1, \dots, \mathbf{P}_M \rrbracket$ and $\mathbb{Q}(\mathbf{Y}) = \mathbf{Y} - \mathbb{P}(\mathbf{Y})$, where $\mathbf{P}_m \in \mathbb{R}^{q_m \times q_m}$ is the projection matrix onto a linear subspace $S_m \subset \mathbb{R}^{q_m}$. We consider the following envelope assumption:

$$\mathbb{Q}(\mathbf{Y}) \mid \mathbf{X} \stackrel{d}{=} \mathbb{Q}(\mathbf{Y}), \quad \mathbb{Q}(\mathbf{Y}) \perp \mathbb{P}(\mathbf{Y}) \mid \mathbf{X}. \tag{39.18}$$

Parallel to the multivariate response envelope model, the envelope assumption in (39.18) indicates that there is a series of possibly low-rank subspaces s_1, \dots, s_M that contain all the information for the estimation of \mathbf{B} . If we identify these subspaces, we can discard their complements without any loss of information.

Moreover, if we set $u_m \leq q_m$ to be the rank of S_m , $\boldsymbol{\Gamma}_m \in \mathbb{R}^{q_m \times u_m}$ to be a basis for S_m and $\boldsymbol{\Gamma}_{0m}$ to be the orthogonal complement of $\boldsymbol{\Gamma}_m$, the envelope assumption leads to the following parsimonious representation of \mathbf{B} and $\boldsymbol{\Sigma}_m$:

$$\mathbf{B} = \llbracket \boldsymbol{\Theta}; \boldsymbol{\Gamma}_1, \dots, \boldsymbol{\Gamma}_M, \mathbf{I}_p \rrbracket, \quad \boldsymbol{\Sigma}_m = \boldsymbol{\Gamma}_m \boldsymbol{\Omega}_m \boldsymbol{\Gamma}_m^T + \boldsymbol{\Gamma}_{0m} \boldsymbol{\Omega}_{0m} \boldsymbol{\Gamma}_{0m}^T, \tag{39.19}$$

where $\boldsymbol{\Theta} \in \mathbb{R}^{u_1 \times \dots \times u_M \times p}$, $\boldsymbol{\Omega}_m \in \mathbb{R}^{u_m \times u_m}$, and $\boldsymbol{\Omega}_{0m} \in \mathbb{R}^{(q_m - u_m) \times (q_m - u_m)}$. As a result, we can extract all the information about \mathbf{B} by focusing on the ‘‘surrogate response’’ $\mathbf{Z} = \llbracket \mathbf{Y}; \boldsymbol{\Gamma}_1^T, \dots, \boldsymbol{\Gamma}_M^T \rrbracket \in \mathbb{R}^{u_1 \times \dots \times u_M}$. With the envelope representation, the number of free parameters reduces from $p \prod_{m=1}^M q_m + \sum_{m=1}^M q_m(q_m + 1)/2$ to $p \prod_{m=1}^M u_m + \sum_{m=1}^M \{u_m(q_m - u_m) + u_m(u_m + 1)/2 + (q_m - u_m)(q_m - u_m + 1)/2\}$, which leads to a reduction of $p(\prod_{m=1}^M q_m - \prod_{m=1}^M u_m)$ free parameters. The smaller number of parameter encourages efficient estimation and computation in the estimation, which we discuss in the next section.

39.3.3 Estimation

The parameters in the tensor response envelope model are estimated by MLE. For the covariance matrices $\boldsymbol{\Sigma}_m$, we let $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_M \otimes \dots \otimes \boldsymbol{\Sigma}_1$. We want to minimize the negative log-likelihood function:

$$\begin{aligned} \ell(\mathbf{B}, \boldsymbol{\Sigma}) &= \log |\boldsymbol{\Sigma}| + \frac{1}{n} \sum_{i=1}^n \left\{ \text{vec}(\mathbf{Y}_i) - \mathbf{B}_{(M+1)}^\top \mathbf{X}_i \right\}^\top \\ &\quad \boldsymbol{\Sigma}^{-1} \left\{ \text{vec}(\mathbf{Y}_i) - \mathbf{B}_{(M+1)}^\top \mathbf{X}_i \right\}. \end{aligned} \quad (39.20)$$

Note that, because of our envelope assumption in (39.19), the objective function $\ell(\mathbf{B}, \boldsymbol{\Sigma})$ implicitly depends on $\boldsymbol{\Gamma}_m$, $\boldsymbol{\Theta}$, $\boldsymbol{\Omega}_m$, $\boldsymbol{\Omega}_{0m}$ as well.

The minimization of (39.20) is performed by minimizing $\ell(\mathbf{B}, \boldsymbol{\Sigma})$ over one parameter while fixing all the others fixed. We start by finding initial estimates for \mathbf{B} and $\boldsymbol{\Sigma}$. We fit each element of the tensor response \mathbf{Y} versus \mathbf{X} to find the OLS estimator \mathbf{B}_{OLS} and use it as the initial value $\mathbf{B}^{(0)}$. For $\boldsymbol{\Sigma}_m^{(0)}$, we employ the following estimator similar to that in [23]:

$$\boldsymbol{\Sigma}_m^{(0)} = \frac{1}{n \prod_{j \neq m} p_j} \sum_{i=1}^n \mathbf{e}_{i(m)} \{ \boldsymbol{\Sigma}_{-m}^{(0)} \}^{-1} \mathbf{e}_{i(m)}, \quad (39.21)$$

where $\mathbf{e}_{i(m)}$ is the mode- m matricization of $\mathbf{e}_i = \mathbf{Y}_i - \mathbf{B}^{(0)} \times_{M+1} \mathbf{X}_i$.

We iterate over m to determine $\boldsymbol{\Sigma}_m^{(0)}$. In the following iterations, it is easy to update \mathbf{B} and $\boldsymbol{\Sigma}$ when $\boldsymbol{\Gamma}_m$, $\boldsymbol{\Theta}$, $\boldsymbol{\Omega}_m$, $\boldsymbol{\Omega}_{0m}$ are fixed, thanks to (39.19). Thus we focus on the update of $\boldsymbol{\Theta}$, $\boldsymbol{\Omega}_m$, $\boldsymbol{\Omega}_{0m}$ at the $(t+1)$ -th iteration. It can be shown that

$$\begin{aligned} \boldsymbol{\Gamma}_m^{(t+1)} &= \arg \min_{\mathbf{G}_m \in \mathbb{R}^{q_m \times u_m}} \log |\mathbf{G}_m^\top \mathbf{M}_m^{(t)} \mathbf{G}_m| \\ &\quad + \log |\mathbf{G}_m^\top (\mathbf{N}_m^{(t)})^{-1} \mathbf{G}_m|, \\ \text{s.t. } \mathbf{G}_m^\top \mathbf{G}_m &= \mathbf{I}_{u_m}, \end{aligned} \quad (39.22)$$

where $\mathbf{M}_m^{(t)} = (n \prod_{j \neq m} q_j)^{-1} \sum_{i=1}^n \boldsymbol{\delta}_{i(m)}^{(t)} \{ \boldsymbol{\Sigma}_{-m}^{(t)} \}^{-1} \boldsymbol{\delta}_{i(m)}^\top$, $\mathbf{N}_m^{(t)} = (n \prod_{j \neq m} q_j)^{-1} \sum_{i=1}^n \mathbf{Y}_{i(m)}^{(t)} \{ \boldsymbol{\Sigma}_{-m}^{(t)} \}^{-1} \mathbf{Y}_{i(m)}^\top$, and $\boldsymbol{\delta}_i^{(t)} = \mathbf{Y}_i - \llbracket \mathbf{B}_{\text{OLS}}; \mathbf{P}_{\Gamma_1}^{(t)}, \dots, \mathbf{P}_{\Gamma_{m-1}}^{(t)}, \mathbf{I}_{q_m}, \mathbf{P}_{\Gamma_{m+1}}^{(t)}, \dots, \mathbf{P}_{\Gamma_M}^{(t)}, \mathbf{I}_p \rrbracket \times_{(M+1)} \mathbf{X}_i^\top$ is the current fitted envelope model residual. The optimization in (39.22) is over all $(q_m \times u_m)$ -dimensional Grassmann manifolds and is the most time-consuming step in our estimation.

With $\{\boldsymbol{\Gamma}_m\}_{m=1}^M$, the parameter $\boldsymbol{\Theta}$ can be estimated by restricting our attention to the regression problem with the surrogate response $\mathbf{Z} = \llbracket \mathbf{Y}; \boldsymbol{\Gamma}_1^\top, \dots, \boldsymbol{\Gamma}_M^\top \rrbracket$ and the predictor \mathbf{X} . More specifically, define $\mathbf{Z}_i^{(t)} = \llbracket \mathbf{Y}_i; (\boldsymbol{\Gamma}_1^{(t+1)})^\top, \dots, (\boldsymbol{\Gamma}_M^{(t+1)})^\top \rrbracket$ and stack them to form $\mathbb{Z}^{(t)} \in \mathbb{R}^{u_1 \times \dots \times u_M \times n}$. Similarly, stack \mathbf{X}_i to form $\mathbb{X} \in \mathbb{R}^{p \times n}$. Our estimate for $\boldsymbol{\Theta}$ is

$$\boldsymbol{\Theta}^{(t+1)} = \mathbb{Z}^{(t)} \times_{(M+1)} \{ (\mathbb{X} \mathbb{X}^\top)^{-1} \mathbb{X} \}. \quad (39.23)$$

Finally, we have

$$\boldsymbol{\Omega}_m^{(t+1)} = \frac{1}{n \prod_{j \neq m} q_j} \sum_{i=1}^n \mathbf{s}_{i(m)}^{(t)} \left\{ \boldsymbol{\Omega}_{-m}^{(t+1)} \right\}^{-1} \mathbf{s}_{i(m)}^\top, \quad (39.24)$$

$$\begin{aligned} \boldsymbol{\Omega}_{0m}^{(t+1)} &= \frac{1}{n \prod_{j \neq m} q_j} \sum_{i=1}^n \left(\boldsymbol{\Gamma}_{0m}^{(t+1)} \right)^\top \mathbf{Y}_{i(m)} \\ &\quad \left\{ \boldsymbol{\Sigma}_{-m}^{(t+1)} \right\}^{-1} \mathbf{Y}_{i(m)}^\top \boldsymbol{\Gamma}_{0m}^{(t+1)}, \end{aligned} \quad (39.25)$$

where $\mathbf{s}_i^{(t)} = \mathbf{Z}_i^{(t)} - \boldsymbol{\Theta}^{(t+1)} \times_{(M+1)} \mathbf{X}_i$.

With these results, the MLE can be found with an iterative algorithm. However, if time becomes an issue, one could consider two modifications to speed up the computation. First, the most challenging step in the optimization is apparently (39.22). One could carry out this step with an efficient 1D algorithm [19] shown to be faster and more stable. Second, instead of iterating over the steps, we can only update the estimates once. As it turns out, whether we iterate until convergence or not, the final estimates are \sqrt{n} -consistent, with smaller asymptotic variance than the OLS estimator.

Finally, we note that the envelope dimensions u_m are typically unknown in practice [91]. We select them individually by minimizing

$$\begin{aligned} \text{BIC}_m &= -\frac{n}{2} \log |\boldsymbol{\Gamma}_m^\top \boldsymbol{\Sigma}_m^{(0)} \boldsymbol{\Gamma}_m| - \frac{n}{2} \log |\boldsymbol{\Gamma}_m^\top (\mathbf{N}_m^{(0)})^{-1} \boldsymbol{\Gamma}_m| \\ &\quad + \log n \cdot p u_m. \end{aligned} \quad (39.26)$$

Since the dimensions are selected marginally, the computation is usually very fast for this aspect.

39.3.4 A Technical Remark

Although seemingly straightforward, the estimation of the tensor response regression is made possible by a striking difference between vector and tensor data. Note that to find the initial value $\boldsymbol{\Sigma}_m^{(0)}$, we have to invert the large covariance $\boldsymbol{\Sigma}_{-m}^{(0)} \in \mathbb{R}^{(\prod_{j \neq m} p_j) \times (\prod_{j \neq m} p_j)}$. It is well-known that, on vector data, we want to avoid inverting the covariance if the dimension is large, because the inverse does not exist when the dimension exceeds the sample size. However, this is not an issue for tensor data even when $\prod_{j \neq m} p_j$ is much larger than the sample size. Note that $\boldsymbol{\Sigma}_{-m}^{(0)}$ is invertible if and only if $\boldsymbol{\Sigma}_j^{(0)}$, $j \neq m$ are all invertible. With some linear algebra we can show that this is true as long as $n \prod_{j \neq m} p_j > p_m$. In most tensor data applications, p_j , $j = 1, \dots, M$ are roughly of the same order. It follows that $\boldsymbol{\Sigma}_j^{(0)}$ is typically invertible, even for very small sample sizes.

Moreover, because of the tensor structure, finding the inverse is generally fast. Recall that $\{ \boldsymbol{\Sigma}_{-m}^{(0)} \}^{-1} = \otimes_{j \neq m} \{ \boldsymbol{\Sigma}_j^{(0)} \}^{-1}$. Hence, there is no need to direct invert the large matrix $\boldsymbol{\Sigma}_{-m}^{(0)}$. Instead, we only need to invert the much smaller matrices $\boldsymbol{\Sigma}_j^{(0)}$. Indeed, this trick is used in many tensor methods that require inverting the separable covariance.

Algorithm 1 The SIMPLS algorithm

1. Initialize $\mathbf{C}_0 = \text{cov}(\mathbf{X}, \mathbf{Y}) \in \mathbb{R}^{p \times q}$.
2. For $j = 1, \dots, u$, do the following:
 - a. Find $\mathbf{w}_j = \arg \max_{\mathbf{w} \in \mathbb{R}^p} \mathbf{w}^T \mathbf{C}_{j-1} \mathbf{C}_{j-1} \mathbf{w}$, s.t. $\mathbf{w}^T \mathbf{w} = 1$;
 - b. Let $u_j = \mathbf{w}_j^T \mathbf{X} \in \mathbb{R}$;
 - c. Compute $\mathbf{v}_j = \text{cov}(\mathbf{X}, t_j) / \text{var}(t_j) \in \mathbb{R}^p$;
 - d. Deflate the cross covariance $\mathbf{C}_j = \mathbf{Q}_j \mathbf{C}_0$, where $\mathbf{Q}_j \in \mathbb{R}^{p \times p}$ is the projection matrix onto the orthogonal subspace of $\text{span}(\mathbf{v}_1, \dots, \mathbf{v}_j)$.
3. Reduce $\mathbf{X} \in \mathbb{R}^p$ to $\mathbf{U} = \mathbf{W}^T \mathbf{X} = (t_1, \dots, t_u) \in \mathbb{R}^u$, where $\mathbf{W} = (\mathbf{w}_1, \dots, \mathbf{w}_u)$.
4. Regress $\mathbf{Y} \in \mathbb{R}^q$ on $\mathbf{U} \in \mathbb{R}^u$.

39.4 Tensor Envelope Partial Least Squares Regression

39.4.1 Partial Least Squares Regression

We first review partial least squares (PLS) for vector data. Consider the response $\mathbf{Y} \in \mathbb{R}^q$ and the predictor $\mathbf{X} \in \mathbb{R}^p$. PLS looks for a reduction of \mathbf{X} in the form of $\mathbf{U} = \mathbf{W}^T \mathbf{X} \in \mathbb{R}^u$ that is used in the follow-up regression as the predictor. Conceptually, PLS attempts to fit a factor model

$$\mathbf{X} = \mathbf{T}\mathbf{U} + \mathbf{E}, \quad \mathbf{Y} = \mathbf{V}\mathbf{U} + \mathbf{F}, \quad (39.27)$$

where $\mathbf{U} \in \mathbb{R}^u$ contains the factors shared by \mathbf{X} and \mathbf{Y} , $\mathbf{T} \in \mathbb{R}^{p \times u}$, $\mathbf{V} \in \mathbb{R}^{q \times u}$ are the loadings, and \mathbf{E}, \mathbf{F} are statistical errors. PLS first recovers the latent factor \mathbf{U} and then estimates the loading \mathbf{V} by fitting a regression model on $\{\mathbf{Y}, \mathbf{U}\}$.

However, unlike many other statistical methods, PLS was first proposed as a heuristic algorithm, without a clear statistical model behind it. Nevertheless, its competitive performance wins its wide popularity. Up until today, many PLS algorithms have been developed [21, 79, e.g.]. We only review one of them named SIMPLS, as it is one of the rare PLS algorithms with a population interpretation [11, 38]. SIMPLS is listed in Algorithm 1. To avoid notation proliferation, we use the parameters such as the cross variance in this algorithm, but of course in practice these parameters are replaced with their sample estimates. SIMPLS sequentially maximizes the correlation between the predictor and the response. It can be shown that on the population level SIMPLS finds an envelope of \mathbf{X} that contains all the information for \mathbf{Y} .

39.4.2 A Tensor PLS Algorithm

Consider a regression problem with $Y \in \mathbb{R}^q$ as the response and $\mathbf{X} \in \mathbb{R}^{p_1 \times \dots \times p_M}$ as the predictor. This is different from the tensor response regression discussed in Sect. 39.3, as now

Algorithm 2 The tensor PLS algorithm

1. Initialize $\mathbf{C}_0 = \text{cov}(\mathbf{X}, \mathbf{Y}) \in \mathbb{R}^{p_1 \times \dots \times p_M \times q}$ and the covariance matrices $\Sigma_{\mathbf{Y}} = \text{cov}(\mathbf{Y})$ and $\Sigma_{\mathbf{X}} = \text{cov}\{\text{vec}(\mathbf{X})\} = \Sigma_M \otimes \dots \otimes \Sigma_1$.
2. For $m = 1, \dots, M$, do the following:
 - a. Standardize the mode- m cross covariance matrix:
$$\tilde{\mathbf{C}}_{0m} = \mathbf{C}_{0(m)} (\Sigma_{\mathbf{Y}}^{-1/2} \otimes \Sigma_{-m}^{-1/2}) \in \mathbb{R}^{p_m \times (q \prod_{k \neq m} p_k)} \quad (39.28)$$
 - b. For $j = 1, \dots, u_m$, do the following
 - i. Find $\mathbf{w}_{mj} = \arg \max_{\mathbf{w} \in \mathbb{R}^{p_m}} \mathbf{w}^T \tilde{\mathbf{C}}_{j-1, m} \tilde{\mathbf{C}}_{j-1, m} \mathbf{w}$, s.t. $\mathbf{w}^T \mathbf{w} = 1$;
 - ii. Let $u_j = \mathbf{w}_j^T \mathbf{X} \in \mathbb{R}$;
 - iii. Compute $\mathbf{v}_j = \text{cov}(\mathbf{X}, t_j) / \text{var}(t_j) \in \mathbb{R}^p$;
 - iv. Deflate the cross covariance $\tilde{\mathbf{C}}_{jm} = \mathbf{Q}_{jm} \tilde{\mathbf{C}}_{0m}$, where $\mathbf{Q}_{jm} \in \mathbb{R}^{p_m \times p_m}$ is the projection matrix onto the orthogonal subspace of $\text{span}(\Sigma_m \mathbf{W}_{jm})$, and $\mathbf{W}_{jm} = (\mathbf{w}_{1m}, \dots, \mathbf{w}_{jm}) \in \mathbb{R}^{p_m \times j}$.
3. Reduce $\mathbf{X} \in \mathbb{R}^{p_1 \times \dots \times p_M}$ to $\mathbf{U} = [\mathbf{X}; \mathbf{W}_1^T, \dots, \mathbf{W}_M^T] \in \mathbb{R}^{u_1 \times \dots \times u_M}$, where $\mathbf{W}_m = (\mathbf{w}_{1m}, \dots, \mathbf{w}_{u_m m}) \in \mathbb{R}^{p_m \times j}$.
4. Regress $\mathbf{Y} \in \mathbb{R}^q$ on $\mathbf{U} \in \mathbb{R}^{u_1 \times \dots \times u_M}$.

the tensor is in the place of the predictor. A natural way to extend the PLS algorithm is to seek a low-rank reduction $\mathbf{U} = [\mathbf{X}; \mathbf{W}_1^T, \dots, \mathbf{W}_M^T]$, where $\mathbf{W}_m \in \mathbb{R}^{p_m \times u_m}$. However, close attention is required to multiple details to ensure a statistical interpretation behind the algorithm. We introduce one such algorithm in Algorithm 2 [89].

We make two remarks concerning Algorithm 2. First, we again use the parameters to avoid notation proliferation, but in practice they should be replaced with estimates. The parameters $\mathbf{C}_0, \Sigma_{\mathbf{Y}}$ are replaced with their usual sample estimates. We explain the more involved estimators for $\Sigma_m, m = 1, \dots, M$ in $\Sigma_{\mathbf{X}}$. By (39.8), The moment-based estimators are

$$\hat{\Sigma}_m = \left(n \prod_{j \neq m} p_j \right)^{-1} \sum_{i=1}^n \mathbf{X}_{i(m)} \mathbf{X}_{i(m)}^T. \quad (39.29)$$

Alternatively, if one strongly believes in the normality assumptions, MLE can be applied [59]. It can be shown that the MLE, $\hat{\Sigma}_m^{\text{MLE}}, m = 1, \dots, M$ must satisfy

$$\hat{\Sigma}_m^{\text{MLE}} = \frac{1}{n \prod_{j \neq m} p_j} \sum_{i=1}^n \mathbf{X}_{i(m)} \left\{ \hat{\Sigma}_{-m}^{\text{MLE}} \right\}^{-1} \mathbf{X}_{i(m)}. \quad (39.30)$$

Hence, the MLEs can be computed iteratively over m .

Second, the tensor PLS algorithm has a similar structure to SIMPLS, but it reduces the tensor predictor along all the modes. Such reduction is important for the sake of both computational and statistical efficiency. Because the reduction is performed along each mode, the number of parameters is under control even when the tensors have a huge number of elements. Moreover, the tensor PLS algorithm turns out to have a population interpretation as well, as we explain in the next section.

39.4.3 A Population Model

Recall that we have a q -dimensional response \mathbf{Y} and a tensor predictor \mathbf{X} . Consider the model

$$Y_k = \langle \mathbf{B}_{::k}, \mathbf{X} \rangle + \epsilon_k, \quad k = 1, \dots, q, \quad (39.31)$$

where $\mathbf{B}_{::k} \in \mathbb{R}^{p_1 \times \dots \times p_M}$ is the coefficient tensor for (Y_k, \mathbf{X}) and ϵ_k is the statistical error independent of \mathbf{X} . We collect the coefficients $\mathbf{B}_{::k}$ to form the coefficient tensor $\mathbf{B} \in \mathbb{R}^{p_1 \times \dots \times p_M \times q}$. The regression model in (39.31) can be equivalently written as

$$\mathbf{Y} = \mathbf{B}_{(M+1)} \text{vec}(\mathbf{X}) + \boldsymbol{\epsilon}, \quad (39.32)$$

where $\boldsymbol{\epsilon} \in \mathbb{R}^q$. We further assume that the covariance of \mathbf{X} is separable in that $\boldsymbol{\Sigma}_{\mathbf{X}} = \text{cov}\{\text{vec}(\mathbf{X})\} = \boldsymbol{\Sigma}_M \otimes \dots \otimes \boldsymbol{\Sigma}_1$. Finally, similar to the tensor response regression model, we assume an envelope structure exists. More specifically, we assume the existence of a series of subspaces $S_m \in \mathbb{R}^{p_m}$, $m = 1, \dots, M$ such that

$$\mathbf{X} \times_m \mathbf{Q}_m \perp \mathbf{X} \times_m \mathbf{P}_m, \quad \mathbf{Y} \perp \mathbf{X} \times_m \mathbf{Q}_m \mid \mathbf{X} \times_m \mathbf{P}_m, \quad (39.33)$$

where $\mathbf{P}_m \in \mathbb{R}^{p_m \times p_m}$ is the projection matrix onto S_m and $\mathbf{Q}_m = \mathbf{I}_{p_m} - \mathbf{P}_m$ is the orthogonal complement. Let $\boldsymbol{\Gamma}_m \in \mathbb{R}^{p_m \times u_m}$ be a basis of S_m and $\boldsymbol{\Gamma}_{0m} \in \mathbb{R}^{p_m \times (p_m - u_m)}$ be a basis of the complement of S_m . The envelope assumption implies that that $\llbracket \mathbf{X}; \mathbf{P}_1, \dots, \mathbf{P}_M \rrbracket$ contains all the information for \mathbf{Y} . Extracting this material part improves the estimation efficiency and lowers the computation cost.

The envelope assumption is integrated into the estimation by noting that

$$\mathbf{B} = \llbracket \boldsymbol{\Theta}; \boldsymbol{\Gamma}_1, \dots, \boldsymbol{\Gamma}_M, \mathbf{I}_q \rrbracket, \quad \text{for some } \boldsymbol{\Theta} \in \mathbb{R}^{u_1 \times \dots \times u_M},$$

$$\boldsymbol{\Sigma}_m = \boldsymbol{\Gamma}_m \boldsymbol{\Omega}_m \boldsymbol{\Gamma}_m^\top + \boldsymbol{\Gamma}_{0m} \boldsymbol{\Omega}_{0m} \boldsymbol{\Gamma}_{0m}^\top, \quad m = 1, \dots, M,$$

where $\boldsymbol{\Omega}_m \in \mathbb{R}^{u_m \times u_m}$, $\boldsymbol{\Omega}_{0m} \in \mathbb{R}^{(p_m - u_m) \times (p_m - u_m)}$ are symmetric positive definite matrices. When u_m 's are small, we achieve noticeable reduction in the number of free parameters.

It can be shown that the tensor PLS algorithm estimates the above model. Let $\mathcal{W}_m \subset \mathbb{R}^{p_m}$ be the subspace spanned by $(\mathbf{w}_{1m}, \dots, \mathbf{w}_{u_m m}) \in \mathbb{R}^{p_m \times u_m}$. As $n \rightarrow \infty$, \mathcal{W}_m converges to S_m . Also, the resulting estimator $\hat{\mathbf{B}}$ is a \sqrt{n} -consistent estimator for \mathbf{B} . This fact provides supports and explanations for the favorable performance of tensor PLS.

39.5 Covariate-Adjusted Tensor Classification in High Dimensions

39.5.1 Linear Discriminant Analysis

The method covariate-adjusted tensor classification in high dimensions (CATCH) is partially motivated by the well-known linear discriminant analysis (LDA). We review some basics for LDA for completeness. For a more comprehensive introduction to LDA, see [60]. Consider the class label $Y = 1, \dots, K$ and the vector predictor $\mathbf{X} \in \mathbb{R}^p$. The LDA model assumes that

$$\Pr(Y = y) = \pi_y, \quad \mathbf{X} \mid Y = y \sim N(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}), \quad (39.34)$$

where $\boldsymbol{\mu}_y \in \mathbb{R}^p$, $\boldsymbol{\Sigma} \in \mathbb{R}^{p \times p}$. The optimal classifier can be explicitly derived under this model, which takes the form

$$\hat{Y} = \arg \max_{k=1, \dots, K} \Pr(Y = k \mid \mathbf{X}) = \arg \max_{k=1, \dots, K} \{a_k + \boldsymbol{\beta}_k^\top \mathbf{X}\}, \quad (39.35)$$

where $\boldsymbol{\beta}_k = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_1)$ and $a_k = \log(\pi_k/\pi_1) - \boldsymbol{\beta}_k^\top(\boldsymbol{\mu}_k + \boldsymbol{\mu}_1)/2$. LDA has been well-received since its introduction, due to its elegant linear form and competitive performance [36, e.g.,]. We note that the linear form is a result of the normality assumption and the homogeneity across classes. In the past a few years, there have been considerable interests in generalizing LDA to high dimensions [7, 10, 27, 54, 58, 65, 78, 81], but these methods are not adapted to tensor classification.

39.5.2 Model

Consider random variables $\{\mathbf{U}, \mathbf{X}, Y\}$, where $\mathbf{U} \in \mathbb{R}^q$ is the vector predictor, $\mathbf{X} \in \mathbb{R}^{p_1 \times \dots \times p_M}$ is the tensor predictor, and $Y = 1, \dots, K$ is the class label. Our goal is to build a classifier for the prediction of Y based on $\{\mathbf{U}, \mathbf{X}\}$. The CATCH model assumes that

$$\Pr(Y = k) = \pi_k, \quad \mathbf{U} \mid (Y = k) \sim N(\boldsymbol{\phi}_k, \boldsymbol{\Psi}), \quad (39.36)$$

$$\mathbf{X} \mid (\mathbf{U} = \mathbf{u}, Y = k) \sim TN(\boldsymbol{\mu}_k + \boldsymbol{\alpha} \bar{\times}_{(M+1)} \mathbf{u}, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_M), \quad (39.37)$$

where $\boldsymbol{\phi}_k \in \mathbb{R}^q$ and $\boldsymbol{\Psi} \in \mathbb{R}^{q \times q}$ are the mean and the covariance of \mathbf{U} within Class k . The parameter $\boldsymbol{\alpha} \in \mathbb{R}^{p_1 \times \dots \times p_M \times q}$ is the effect of \mathbf{U} on \mathbf{X} within class. After we adjust \mathbf{X} for \mathbf{U} , its within-class mean and variance are given by $\boldsymbol{\mu}_k \in \mathbb{R}^{p_1 \times \dots \times p_M}$, and $\boldsymbol{\Sigma}_m \in \mathbb{R}^{p_m \times p_m}$, $m = 1, \dots, M$, respectively.

The CATCH model is similar to the LDA model in the assumption of normality and constant covariance. Indeed, if we ignore \mathbf{X} , then (Y, \mathbf{U}) follows the LDA model. However, in the CATCH model there are two layers of dependence, which allows us to model the relationship between \mathbf{X} and \mathbf{U} as well. Within Class $Y = k$, \mathbf{X} and \mathbf{U} are connected with a tensor response regression model with \mathbf{X} as the response and \mathbf{U} as the predictor; see Sect. 39.3. Moreover, in the absence of \mathbf{U} , \mathbf{X} is assumed to follow the tensor normal distribution:

$$\mathbf{X} | Y = k \sim TN(\boldsymbol{\mu}_k; \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_M). \quad (39.38)$$

This can be viewed as a model for tensor discriminant analysis (TDA) that generalizes LDA to tensor data. The tensor normal distribution reduces the number of free parameters in the covariance of \mathbf{X} .

Under the CATCH model, we derive the optimal classifier to be

$$\begin{aligned} \hat{Y} &= \arg \max_{k=1, \dots, K} \Pr(Y = k | \mathbf{X}, \mathbf{U}) \\ &= \arg \max_{k=1, \dots, K} \{a_k + \boldsymbol{\gamma}_k^\top \mathbf{U} + \langle \mathbf{B}_k, \mathbf{X} - \boldsymbol{\alpha} \bar{\mathbf{x}}_{(M+1)} \mathbf{U} \rangle\}, \end{aligned} \quad (39.39)$$

where $\boldsymbol{\gamma}_k = \boldsymbol{\Psi}^{-1}(\boldsymbol{\phi}_k - \boldsymbol{\phi}_1)$, $\mathbf{B}_k = \llbracket \boldsymbol{\mu}_k - \boldsymbol{\mu}_1; \boldsymbol{\Sigma}_1^{-1}, \dots, \boldsymbol{\Sigma}_M^{-1} \rrbracket$, and $a_k = \log(\pi_k/\pi_1) - \frac{1}{2} \boldsymbol{\gamma}_k^\top (\boldsymbol{\phi}_k + \boldsymbol{\phi}_1) - \langle \mathbf{B}_k, \frac{1}{2}(\boldsymbol{\mu}_k + \boldsymbol{\mu}_1) \rangle$ is a scalar that does not involve \mathbf{X} or \mathbf{U} . The coefficient $\boldsymbol{\gamma}_k$ is the direct effect of \mathbf{U} on Y , while $\boldsymbol{\alpha}$ is the indirect effect of \mathbf{U} on Y through \mathbf{X} . The coefficients \mathbf{B}_k summarizes the net effect of \mathbf{X} on Y after it is adjusted for \mathbf{U} . We demonstrate the structure of our CATCH model in Fig. 39.4.

So far we only utilize the tensor structure of \mathbf{X} by making the tensor normal assumption. When the tensor is large, more efforts may be needed for parsimonious modeling. We achieve this goal by assuming that the coefficients \mathbf{B}_k are sparse. Note that, for any $j_1 \cdots j_M$, the coefficients $b_{2, j_1 \cdots j_M}, \dots, b_{K, j_1 \cdots j_M}$ are related to the same element $x_{j_1 \cdots j_M}$. They are the effects of the same element in distinguishing

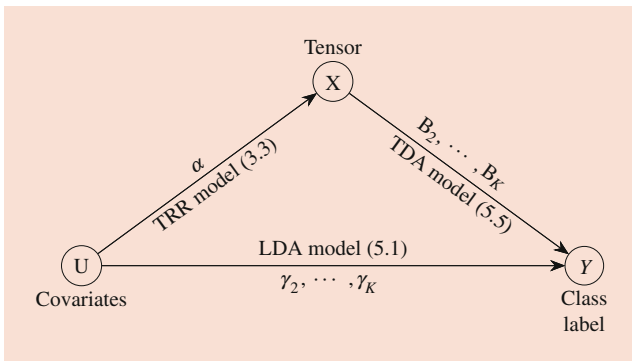


Fig. 39.4 Graphical illustration of the CATCH model. TRR stands for tensor response regression. A similar demonstration was presented in the original proposal of CATCH [61] as well

between different pairs of classes. It follows that the sparsity in \mathbf{B}_k has a group structure:

$$\begin{aligned} x_{j_1 \cdots j_M} \text{ is not important for classification} &\Leftrightarrow b_{2, j_1 \cdots j_M} \\ &= \dots = b_{K, j_1 \cdots j_M} = 0. \end{aligned} \quad (39.40)$$

We will incorporate such sparsity assumption in estimation.

39.5.3 Estimation

Suppose that we collect n observations $\{\mathbf{X}_i, \mathbf{U}_i, Y_i\}_{i=1}^n$. We construct estimators of the parameters in the optimal classifier (39.39). The estimators for $\{\pi_k, \boldsymbol{\phi}_k, \boldsymbol{\gamma}_k, \boldsymbol{\mu}_k, \mathbf{B}_k\}$ can be easily found by maximizing the likelihood function, although we need more notation to ease the presentation. For the observations with $Y_i = k$, let $\tilde{\mathbf{X}}_i = \mathbf{X}_i - \bar{\mathbf{X}}_k$ and $\tilde{\mathbf{U}}_i = \mathbf{U}_i - \bar{\mathbf{U}}_k$ be the centered tensor and vector predictors. We further stack these centered predictors to form $\tilde{\mathbf{X}} \in \mathbb{R}^{p_1 \times \dots \times p_M \times n}$ and $\tilde{\mathbf{U}} \in \mathbb{R}^{q \times n}$, respectively. Straightforward calculation shows that the MLEs are

$$\hat{\pi}_k = \frac{1}{n} \sum_{i=1}^n 1(Y_i = k), \quad \hat{\boldsymbol{\phi}}_k = \bar{\mathbf{U}}_k, \quad \hat{\boldsymbol{\gamma}}_k = \hat{\boldsymbol{\Psi}}^{-1}(\hat{\boldsymbol{\phi}}_k - \hat{\boldsymbol{\phi}}_1), \quad k = 1, \dots, K,$$

$$\hat{\boldsymbol{\alpha}} = \tilde{\mathbf{X}} \times_{(M+1)} \{(\tilde{\mathbf{U}} \tilde{\mathbf{U}}^\top)^{-1} \tilde{\mathbf{U}}\},$$

$$\text{where } \hat{\boldsymbol{\Psi}} = \frac{1}{n} \sum_{k=1}^K \sum_{Y_i=k} (\mathbf{U}_i - \hat{\boldsymbol{\phi}}_k)(\mathbf{U}_i - \hat{\boldsymbol{\phi}}_k)^\top.$$

The estimation for \mathbf{B}_k requires more attention, though, as we want to impose sparsity. A most popular way to obtain sparse estimates is to apply penalization. But to use penalty functions, we first need to find an estimating equation for \mathbf{B}_k . We note that, on the population level, $\mathbf{B}_2, \dots, \mathbf{B}_K$ are the solution to the following optimization problem:

$$\begin{aligned} (\mathbf{B}_2, \dots, \mathbf{B}_K) &= \arg \min_{\mathbf{B}_2, \dots, \mathbf{B}_K} \sum_{k=2}^K \{ \langle \mathbf{B}_k, \llbracket \mathbf{B}_k; \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_M \rrbracket \rangle \\ &\quad - 2 \langle \mathbf{B}_k, \boldsymbol{\mu}_k - \boldsymbol{\mu}_1 \rangle \}. \end{aligned} \quad (39.41)$$

Equation (39.41) can be used as an estimating equation as long as we find estimates of $\boldsymbol{\Sigma}_m$ and $\boldsymbol{\mu}_k$. By the properties of the tensor normal distribution in Sect. 39.1.4, we construct the following moment-based estimators:

$$\begin{aligned} \hat{\boldsymbol{\mu}}_k &= \bar{\mathbf{X}}_k - \hat{\boldsymbol{\alpha}} \bar{\mathbf{x}}_{(M+1)} \bar{\mathbf{U}}_k, \quad k = 1, \dots, K, \\ \hat{\boldsymbol{\Sigma}}_j &= \tilde{s}_{j,11}^{-1} \tilde{\mathbf{S}}_j \text{ for } j = 1, \dots, M-1; \quad \hat{\boldsymbol{\Sigma}}_M = \frac{\widehat{\text{var}}(X_{1\dots 1})}{\prod_{j=1}^M \tilde{s}_{j,11}} \tilde{\mathbf{S}}_M, \end{aligned}$$

where $\tilde{\mathbf{S}}_j = (n \prod_{l \neq j} p_l)^{-1} \sum_{i=1}^n \hat{\mathbf{E}}_{(j)}^i (\hat{\mathbf{E}}_{(j)}^i)^\top$ with $\hat{\mathbf{E}}_{(j)}^i = \mathbf{X}^i - \hat{\boldsymbol{\mu}}_k - \hat{\boldsymbol{\alpha}} \bar{\mathbf{x}}_{(M+1)} \mathbf{U}^i = (\mathbf{X}^i - \bar{\mathbf{X}}_k) - \hat{\boldsymbol{\alpha}} \bar{\mathbf{x}}_{(M+1)} (\mathbf{U}^i - \bar{\mathbf{U}}_k)$. We

Table 39.1 Classification errors on the ADHD datasets. Reported error rates are averages based on 100 replicates, and standard errors are in parentheses. CP-GLM is not applicable for multiclass problem because multinomial logistic has not been implemented

	CATCH		CP-GLM		ℓ_1 -logistic regression		ℓ_1 -SVM	
Binary	22.79	(0.2)	25.05	(0.19)	23.99	(0.16)	27.54	(0.31)
Multiclass	35.22	(0.25)	NA		35.66	(0.21)	41.28	(0.32)

plug in these estimators and add the group-Lasso penalty to (39.41). The parameters \mathbf{B}_k are estimated by

$$\min_{\mathbf{B}_2, \dots, \mathbf{B}_K} \left[\sum_{k=2}^K \langle \mathbf{B}_k, [\mathbf{B}_k; \hat{\Sigma}_1, \dots, \hat{\Sigma}_M] \rangle - 2 \langle \mathbf{B}_k, \hat{\mu}_k - \hat{\mu}_1 \rangle \right. \\ \left. + \lambda \sum_{j_1 \dots j_M} \sqrt{\sum_{k=2}^K b_{k, j_1 \dots j_M}^2} \right], \quad (39.42)$$

The problem in (39.42) can be solved by a blockwise coordinate descent algorithm where in each step we soft-threshold $b_{k, j_1 \dots j_M}$, $k = 2, \dots, K$. This algorithm is implemented in the R package `catch` available on CRAN. It can be shown that, with a probability of 1, the blockwise coordinate descent algorithm converges to the global minimizer.

The CATCH classifier is also supported by theoretical results. When $\log p_m = o(n)$, under mild conditions the CATCH classifier converges to the optimal classifier. The classification error of CATCH converges to the lowest classification error possible as well.

39.6 A Real Data Example

We demonstrate the performance of CATCH and CP-GLM on a real dataset regarding the attention-deficit hyperactive disorder (ADHD). The ADHD dataset is shared by Neuru Bureau at https://fcon_1000.projects.nitrc.org/indi/adhd200/. The results in this section were originally reported by Pan et al. [61].

The ADHD dataset contains complete rs-fMRI and s-MRI data of 930 subjects. These subjects are divided into four categories: typically developing children (TDC), ADHD combined, ADHD hyperactive, and ADHD inattentive. In addition to the images, age, gender, and handedness are also recorded. Because of the apparent difference between genders, we stratify the data according to the gender of each subject. The MRI data are treated as the tensor predictor (\mathbf{X}), while age and handedness are treated as the vector predictor (\mathbf{U}). The MRI data are preprocessed to be of size $24 \times 27 \times 24$. Two classification problems are considered: a binary problem where we predict whether a subject shows hyperactivity and a three-class problem where subjects are classified into typically developing children (TDC), ADHD combined/hyperactive, and ADHD inattentive.

We repeatedly form training sets of 762 subjects and testing sets of 168 subjects. Two well-known classifiers, ℓ_1 -logistic regression [31, 33] and ℓ_1 -support vector machine (SVM) [2, 4, 20, 22, 32], are also included as baselines. The results are listed in Table 39.1. The tensor methods are very competitive, yielding similar or better accuracies than the baselines.

References

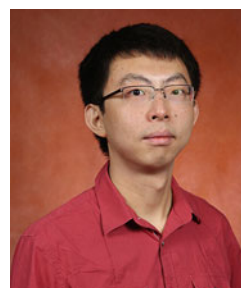
- Bao, Y.T., Chien, J.T.: Tensor classification network. In: 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP), pp. 1–6 (2015)
- Becker, N., Werft, W., Toedt, G., Lichter, P., Benner, A.: Penalizedsvm: a r-package for feature selection svm classification. *Bioinformatics* **25**(13), 1711–1712 (2009)
- Bi, X., Qu, A., Shen, X.: Multilayer tensor factorization with applications to recommender systems. arXiv preprint arXiv:1711.01598 (2017)
- Bradley, P.S., Mangasarian, O.L.: Feature selection via concave minimization and support vector machines. *ICML* **98**, 82–90 (1998)
- Bro, R.: Multiway calibration. *multilinear pls. J. Chemom.* **10**(1), 47–61 (1996)
- Bura, E., Cook, R.D.: Extending sliced inverse regression: The weighted chi-squared test. *J. Am. Stat. Assoc.* **96**, 996–1003 (2001)
- Cai, T., Liu, W.: A direct estimation approach to sparse linear discriminant analysis. *J. Am. Stat. Assoc.* **106**(496), 1566–1577 (2011)
- Carroll, J.D., Chang, J.-J.: Analysis of individual differences in multidimensional scaling via an n-way generalization of “eckart-young” decomposition. *Psychometrika* **35**(3), 283–319 (1970)
- Chi, E.C., Kolda, T.G.: On tensors, sparsity, and nonnegative factorizations. *SIAM J. Matrix Anal. Appl.* **33**(4), 1272–1299 (2012)
- Clemmensen, L., Hastie, T., Witten, D., Ersbøll, B.: Sparse discriminant analysis. *Technometrics* **53**(4), 406–413 (2011)
- Cook, R., Helland, I., Su, Z.: Envelopes and partial least squares regression. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **75**(5), 851–877 (2013)
- Cook, R.D.: Principal components, sufficient dimension reduction, and envelopes. *Annu. Rev. Stat. Appl.* **5**(1), 533–559 (2018)
- Cook, R.D., Li, B., Chiaromonte, F.: Envelope models for parsimonious and efficient multivariate linear regression. *Stat. Sin.* **20**(3), 927–960 (2010)
- Cook, R.D., Ni, L.: Sufficient dimension reduction via inverse regression: A minimum discrepancy approach. *J. Am. Stat. Assoc.* **100**(470), 410–428 (2005)
- Cook, R.D., Weisberg, S.: Comment: Sliced inverse regression for dimension reduction. *J. Am. Stat. Assoc.* **86**(414), 328–332 (1991)
- Cook, R.D., Zhang, X.: Fused estimators of the central subspace in sufficient dimension reduction. *J. Am. Stat. Assoc.* **109**(506), 815–827 (2014)

17. Cook, R.D., Zhang, X.: Foundations for envelope models and methods. *J. Am. Stat. Assoc.* **110**(510), 599–611 (2015)
18. Cook, R.D., Zhang, X.: Simultaneous envelopes for multivariate linear regression. *Technometrics* **57**(1), 11–25 (2015)
19. Cook, R.D., Zhang, X.: Algorithms for envelope estimation. *J. Comput. Graph. Stat.* **25**(1), 284–300 (2016)
20. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)
21. De Jong, S.: Simpls: an alternative approach to partial least squares regression. *Chemom. Intell. Lab. Syst.* **18**(3), 251–263 (1993)
22. Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D., Weingessel, A.: E1071: Misc functions of the department of statistics (e1071), TU Wien. In: CRAN R package, vol. 11 (2009)
23. Dutilleul, P.: The MLE algorithm for the matrix normal distribution. *J. Stat. Comput. Simul.* **64**(2), 105–123 (1999)
24. Eliseyev, A., Aksenova, T.: Recursive n-way partial least squares for brain-computer interface. *PloS One* **8**(7), e69962 (2013)
25. Fan, J., Fan, Y.: High dimensional classification using features annealed independence rules. *Ann. Stat.* **36**(6), 2605–2637 (2008)
26. Fan, J., Feng, Y., Song, R.: Nonparametric independence screening in sparse ultra-high-dimensional additive models. *J. Amer. Statist. Assoc.* **106**, 544–557 (2011)
27. Fan, J., Feng, Y., Tong, X.: A road to classification in high dimensional space: the regularized optimal affine discriminant. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **74**(4), 745–771 (2012)
28. Fan, J., Li, R.: Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Am. Stat. Assoc.* **96**(456), 1348–1360 (2001)
29. Fan, J., Lv, J.: Sure independence screening for ultra-high dimensional feature space. *J. R. Statist. Soc. B* **20**, 101–148 (2008)
30. Fosdick, B.K., Hoff, P.D.: Separable factor analysis with applications to mortality data. *Ann. Appl. Stat.* **8**(1), 120–147 (2014)
31. Friedman, J., Hastie, T., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* **33**(1), 1–22 (2010)
32. Fung, G.M., Mangasarian, O.L.: A feature selection newton method for support vector machine classification. *Comput. Optim. Appl.* **28**(2), 185–202 (2004)
33. Goeman, J., Meijer, R., Chaturvedi, N.: penalized: L1 (lasso and fused lasso) and L2 (ridge) penalized estimation in GLMs and in the cox model. In: CRAN R package (2012)
34. Goldsmith, J., Huang, L., Crainiceanu, C.M.: Smooth scalar-on-image regression via spatial bayesian variable selection. *J. Comput. Graph. Stat.* **23**(1), 46–64 (2014)
35. Gupta, A.K., Nagar, D.K.: *Matrix Variate Distributions*, vol. 104. CRC Press, New York (1999)
36. Hand, D.J.: Classifier technology and the illusion of progress. *Stat. Sci.* **21**(1), 1–14 (2006)
37. Harshman, R.A., et al. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis (1970)
38. Helland, I.S.: On the structure of partial least squares regression. *Commun. Stat. Simul. Comput.* **17**(2), 581–607 (1988)
39. Hoff, P.D.: Separable covariance arrays via the Tucker product, with applications to multivariate relational data. *Bayesian Anal.* **6**(2), 179–196 (2011)
40. Hoff, P.D.: Multilinear tensor regression for longitudinal relational data. *Ann. Appl. Stat.* **9**(3), 1169–1193 (2015)
41. Kiers, H.A.: A three-step algorithm for candecomp/parafac analysis of large data sets with multicollinearity. *Journal of Chemometrics: A Journal of the Chemometrics Society* **12**(3), 155–171 (1998)
42. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. *SIAM Rev.* **51**(3), 455–500 (2009)
43. Lai, Z., Xu, Y., Yang, J., Tang, J., Zhang, D.: Sparse tensor discriminant analysis. *IEEE Trans. Image Process.* **22**(10), 3904–3915 (2013)
44. Lazar, N.: *The Statistical Analysis of Functional MRI Data*. Springer, Berlin (2008)
45. Li, B., Kim, M.K., Altman, N.: On dimension folding of matrix- or array-valued statistical objects. *Ann. Stat.* **38**(2), 1094–1121 (2009)
46. Li, B., Wang, S.: On directional regression for dimension reduction. *J. Am. Stat. Assoc.* **102**, 997–1008 (2007)
47. Li, G., Peng, H., Zhang, J., Zhu, L.-X.: Robust rank correlation based screening. *Ann. Statist.* **40**, 1846–1877 (2012)
48. Li, K.-C.: Sliced inverse regression for dimension reduction. *J. Am. Stat. Assoc.* **86**(414), 316–327 (1991)
49. Li, L., Zhang, X.: Parsimonious tensor response regression. *J. Am. Stat. Assoc.* **112**(519), 1131–1146 (2017)
50. Li, Q., Schonfeld, D.: Multilinear discriminant analysis for higher-order tensor data classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **36**(12), 2524–2537 (2014)
51. Li, R., Zhong, W., Zhu, L.-P.: Feature screening via distance correlation learning. *J. Am. Statist. Assoc.* **107**, 1129–1139 (2012)
52. Liu, T., Yuan, M., Zhao, H.: Characterizing spatiotemporal transcriptome of human brain via low rank tensor decomposition. *arXiv preprint arXiv:1702.07449* (2017)
53. Lock, E.F.: Tensor-on-tensor regression. *J. Comput. Graph. Stat.* **27**(3), 638–647 (2017)
54. Mai, Q., Yang, Y., Zou, H.: Multiclass sparse discriminant analysis. *Stat. Sin.* **29**(1), 97–111 (2019)
55. Mai, Q., Zou, H.: The Kolmogorov filter for variable screening in high-dimensional binary classification. *Biometrika* **100**, 229–234 (2013)
56. Mai, Q., Zou, H.: Nonparametric variable transformation in sufficient dimension reduction. *Technometrics* **57**, 1–10 (2015)
57. Mai, Q., Zou, H.: The fused Kolmogorov filter: A nonparametric model-free screening method. *Ann. Statist.* **43**, 1471–1497 (2015)
58. Mai, Q., Zou, H., Yuan, M.: A direct approach to sparse discriminant analysis in ultra-high dimensions. *Biometrika* **99**(1), 29–42 (2012)
59. Manceur, A.M., Dutilleul, P.: Maximum likelihood estimation for the tensor normal distribution: algorithm, minimum sample size, and empirical bias and dispersion. *J. Comput. Appl. Math.* **239**, 37–49 (2013)
60. Mardia, K., Kent, J., Bibby, J.: *Multivariate analysis*. In: *Probability and Mathematical Statistics*. Academic Press Inc, New Year (1979)
61. Pan, Y., Mai, Q., Zhang, X.: Covariate-adjusted tensor classification in high dimensions. *J. Am. Stat. Assoc.* **114**(527), 1305–1319 (2019)
62. Rajih, M., Comon, P., Harshman, R.A.: Enhanced line search: A novel method to accelerate parafac. *SIAM J. Matrix Anal. Appl.* **30**(3), 1128–1147 (2008)
63. Raskutti, G., Yuan, M.: Convex regularization for high-dimensional tensor regression. *arXiv preprint arXiv:1512.01215* (2015)
64. Reiss, P., Ogden, R.: Functional generalized linear models with images as predictors. *Biometrics* **66**, 61–69 (2010)
65. Shao, J., Wang, Y., Deng, X., Wang, S.: Sparse linear discriminant analysis by thresholding for high dimensional data. *Ann. Stat.* **39**(2), 1241–1265 (2011)
66. Su, Z., Cook, R.D.: Inner envelopes: efficient estimation in multivariate linear regression. *Biometrika* **99**(3), 687–702 (2012)
67. Su, Z., Cook, R.D.: Estimation of multivariate means with heteroscedastic errors using envelope models. *Stat. Sin.* **23**(1), 213–230 (2013)
68. Sun, W.W., Li, L.: Store: sparse tensor response regression and neuroimaging analysis. *J. Mach. Learn. Res.* **18**(1), 4908–4944 (2017)

69. Sun, W.W., Li, L.: Dynamic tensor clustering. *J. Am. Stat. Assoc.* **114**(528), 1894–1907 (2019)
70. Sun, W.W., Lu, J., Liu, H., Cheng, G.: Provable sparse tensor decomposition. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **79**(3), 899–916 (2016)
71. Tao, D., Li, X., Wu, X., Maybank, S.J.: General tensor discriminant analysis and gabor features for gait recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(10), 1700–1715 (2007)
72. Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B* **58**(1), 267–288 (1996)
73. Tomasi, G.: Use of the properties of the khatri-rao product for the computation of Jacobian. In: *Hessian, and Gradient of the PARAFAC Model Under MATLAB* (2005)
74. Tomasi, G.: Practical and computational aspects in chemometric data analysis. In: *Center for Skov, Landskab og Planlægning/Københavns Universitet* (2006)
75. Wang, M., Fischer, J., Song, Y.S., et al.: Three-way clustering of multi-tissue multi-individual gene expression data using seminonnegative tensor decomposition. *Ann. Appl. Stat.* **13**(2), 1103–1127 (2019)
76. Wang, X., Nan, B., Zhu, J., Koeppe, R.: Regularized 3d functional regression for brain image data via haar wavelets. *Ann. Appl. Stat.* **8**(2), 1045 (2014)
77. Wang, X., Zhu, H.: Generalized scalar-on-image regression models via total variation. *J. Am. Stat. Assoc.* **112**(519), 1156–1168 (2017)
78. Witten, D.M., Tibshirani, R.: Penalized classification using fisher's linear discriminant. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **73**(5), 753–772 (2011)
79. Wold, H.: Estimation of principal components and related models by iterative least squares. In: *Multivariate Analysis*, pp. 391–420 (1966)
80. Worsley, K.J., Taylor, J.E., Tomaiuolo, F., Lerch, J.: Unified univariate and multivariate random field theory. *Neuroimage* **23**, S189–S195 (2004)
81. Xu, P., Zhu, J., Zhu, L., Li, Y.: Covariance-enhanced discriminant analysis. *Biometrika* **102**(1), 33–45 (2015)
82. Yan, S., Xu, D., Yang, Q., Zhang, L., Tang, X., Zhang, H.-J.: Discriminant analysis with tensor representation. In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pp. 526–532 (2005)
83. Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **68**(1), 49–67 (2006)
84. Yuan, M., Zhang, C.-H.: On tensor completion via nuclear norm minimization. *Found. Comput. Math.* **16**(4), 1031–1068 (2016)
85. Zeng, R., Wu, J., Senhadji, L., Shu, H.: Tensor object classification via multilinear discriminant analysis network. In: *Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1971–1975 (2015)
86. Zhang, A., et al.: Cross: Efficient low-rank tensor completion. *Ann. Stat.* **47**(2), 936–964 (2019)
87. Zhang, A., Xia, D.: Tensor SVD: Statistical and computational limits. *IEEE Trans. Inf. Theory* **64**(11), 7311–7338 (2018)
88. Zhang, C.: Nearly unbiased variable selection under minimax concave penalty. *Ann. Statist.* **38**, 894–942 (2010)
89. Zhang, X., Li, L.: Tensor envelope partial least-squares regression. *Technometrics* **59**(4), 426–436 (2017)
90. Zhang, X., Mai, Q.: Efficient integration of sufficient dimension reduction and prediction in discriminant analysis. *Technometrics*, accepted (2018)
91. Zhang, X., Mai, Q.: Model-free envelope dimension selection. *Electronic Journal of Statistics* **12**(2), 2193–2216 (2018)
92. Zhao, J., Leng, C.: Structured lasso for regression with matrix covariates. *Stat. Sin.* **24**(2), 799–814 (2014)
93. Zhao, Q., Caiafa, C.F., Mandic, D.P., Chao, Z.C., Nagasaka, Y., Fujii, N., Zhang, L., Cichocki, A.: Higher order partial least squares (HOPLS): a generalized multilinear regression method. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(7), 1660–1673 (2012)
94. Zhong, W., Suslick, K.S.: Matrix discriminant analysis with application to colorimetric sensor array data. *Technometrics* **57**(4), 524–534 (2015)
95. Zhou, H., Li, L.: Regularized matrix regression. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **76**(2), 463–483 (2014)
96. Zhou, J., He, X.: Dimension reduction based on constrained canonical correlation and variable filtering. *Ann. Stat.* **36**, 1649–1668 (2008)
97. Zhu, H., Fan, J., Kong, L.: Spatially varying coefficient model for neuroimaging data with jump discontinuities. *J. Am. Stat. Assoc.* **109**(507), 1084–1098 (2014)
98. Zhu, L.-P., Li, L., Li, R., Zhu, L.-X.: Model-free feature screening for ultrahigh dimensional data. *J. Am. Statist. Assoc.* **106**, 1464–1475 (2011)
99. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **67**(2), 301–320 (2005)



Qing Mai is Associate Professor, Department of Statistics, Florida State University. She received her BS degree in Statistics from Nankai University, China, in 2008, and PhD degree in Statistics from University of Minnesota in 2013. She is a member of the American Statistical Association, the Institute of Mathematical Statistics, and the International Chinese Statistical Association, and is currently serving as Associate Editor of *Biometrics*. Dr. Mai's current research areas include classification, clustering analysis, high-dimensional statistics and inference, tensor data analysis, and machine learning. Background on these works can be found at <https://sites.google.com/view/qingmai/research/>.



Xin (Henry) Zhang is Associate Professor, Department of Statistics, Florida State University. He received his BS degree in Applied Physics from University of Science and Technology of China (USTC) in 2008, and PhD degree in Statistics from University of Minnesota in 2014. He is currently serving as Associate Editor of *Biometrics*. He is a member of the American Statistical Association (ASA), the Institute of Mathematical Statistics (IMS), and the International Chinese Statistical Association (ICSA). He received the Outstanding Young Researcher Award from the ICSA in 2020. His current research areas include dimension reduction, multivariate regression and classification, clustering analysis, highdimensional statistics and inference, network data, tensor data analysis, and biostatistical applications. Background on these works can be found at <https://ani.stat.fsu.edu/~henry/>.



Random Forests for Survival Analysis and High-Dimensional Data

40

Sarah E. Formentini , Yifan Cui , and Ruoqing Zhu

Contents

40.1	Right-Censored Data and Survival Analysis	831
40.1.1	Survival Estimation.....	832
40.1.2	Parametric and Semi-Parametric Models for Survival Data.....	832
40.2	Tree-Based Models	833
40.2.1	Splitting Mechanism.....	834
40.2.2	Tree Prediction.....	836
40.3	Random Forests for Right-Censored Survival Data	836
40.3.1	An Ensemble of Trees with Randomness.....	836
40.3.2	Ensemble Prediction.....	837
40.3.3	Tuning Parameters.....	837
40.3.4	Evaluation Criteria for Survival Models.....	838
40.3.5	Out-of-Bag Error.....	839
40.4	Variable Selection via Variable Importance in High-Dimensional Settings	839
40.4.1	Other Methods for Calculating Variable Importance.....	840
40.5	Example: Breast Cancer Data Analysis	841
40.5.1	Data Processing and Description.....	841
40.5.2	Fitting Survival Random Forest.....	841
40.6	Bias and Corrections of Survival Forests	842
40.6.1	Biasedness of Survival Forests.....	842
40.6.2	A Motivating Example.....	842
40.6.3	Some Solutions for Bias Correction.....	843
40.7	Theoretical Properties of Random Forest	844
40.8	Supplementary Material	845
	References	845

Abstract

One of the most commonly encountered problems in biomedical studies is analyzing censored survival data. Survival analysis differs from standard regression problems by one central feature: the event of interest may not be fully observed. Therefore, statistical methods used to analyze this data must be adapted to handle the missing information. In this chapter, we provide a brief introduction of right-censored survival data and introduce survival random forest models for analyzing them. Random forests are among the most popular machine learning algorithms. During the past decade, they have seen tremendous success in biomedical studies for prediction and decision-making. In addition to the statistical formulation, we also provide details of tuning parameters commonly considered in practice. An analysis example of breast cancer relapse free survival data is used as a demonstration. We further introduce the variable importance measure that serves as a variable selection tool in high-dimensional analysis. These examples are carried out using a newly developed R package RLT, which is available on GitHub.

Keywords

Random forests · Survival analysis · Random survival forest · Tree estimator · Right censoring · High-dimensional data · Variable importance · Variable selection · C-index · Brier score

40.1 Right-Censored Data and Survival Analysis

Survival analysis focuses mainly on analyzing time-to-event data. However, the event times may not be fully observed due to losing a subject to follow-up, the study ending before all

S. E. Formentini · R. Zhu (✉)
University of Illinois at Urbana-Champaign, Champaign, IL, USA
e-mail: sarahef2@illinois.edu; rqzhu@illinois.edu

Y. Cui
Zhejiang University, Hangzhou, Zhejiang, China
e-mail: cuiyf@zju.edu.cn

subjects experience the event, or due to other reasons. This information loss is referred to as censoring. In this chapter, we consider only the most commonly encountered censoring type: right censoring. To be specific, assume that each subject is waiting to experience a failure event, with the (random) waiting time denoted as T . T is also commonly referred to as the failure time. The subject may also be at risk of being lost to follow-up, which would happen at a (random) censoring time C . When C happens before T , we have a right-censoring scenario in the sense that the failure time is at the right-hand side of the censoring time. Hence, the failure time T cannot be fully observed, and we observe C instead.

Following standard notations in survival analysis literature, e.g., [1], let $\mathcal{D}_n = \{\mathbf{x}_i, y_i, \delta_i\}_{i=1}^n$ be a set of n independently and identically distributed (i.i.d.) observed copies of the triplet $\{\mathbf{X}, Y, \Delta\}$, where \mathbf{X} is a p -dimensional random covariate vector, Y is the observed outcome defined as $Y = \min(T, C)$, where T is the random time until the event of interest and C is the random time until the subject is lost to follow-up. We only observe the shorter of the two times. Lastly, $\Delta = \mathbb{1}(T \leq C)$ is a censoring indicator, which indicates if subject i experienced the event of interest or was lost to follow-up. Note that both T and C may depend on the covariate vector \mathbf{X} and, therefore, so do Y and Δ . Using the observed data \mathcal{D}_n , our main goal is to estimate the conditional survival function of T , defined as

$$S(t, \mathbf{x}) = P(T > t | \mathbf{X} = \mathbf{x}). \quad (40.1)$$

The survival function is also one minus the cumulative distribution function $F(t, \mathbf{x})$, defined as $P(T \leq t | \mathbf{X} = \mathbf{x})$. Similar notations are used for other quantities in the remaining parts of this section. Two important related concepts are the cumulative hazard function (CHF), defined as

$$\Lambda(t, \mathbf{x}) = -\log\{S(t, \mathbf{x})\}, \quad (40.2)$$

and the hazard function, defined as

$$\lambda(t, \mathbf{x}) = d\Lambda(t, \mathbf{x})/dt. \quad (40.3)$$

It is then easy to see that knowing any of the above quantities will allow us to calculate the rest. It should also be noted that, for identifiability issues [1], we often assume conditional independence ($T \perp C | \mathbf{X}$), which essentially states that given the covariate values \mathbf{X} , the failure time and censoring time do not provide any additional information to each other. An example that violates this assumption is when C could, with a certain probability, happen at $T - q$ for some unknown value q . In that case, we will not be able to estimate the distribution of T . For information regarding statistical properties and assumptions of survival analysis models, we refer to [1].

40.1.1 Survival Estimation

To facilitate later arguments, we introduce two methods for estimating a survival function given the data $\{y_i, \delta_i\}_{i=1}^n$. First, a Nelson-Aalen estimator [2, 3] for the cumulative hazard function is defined as

$$\hat{\Lambda}(t) = \sum_{k:t_k \leq t} \frac{\sum_{i=1}^n \mathbb{1}(\delta_i = 1) \mathbb{1}(y_i = t_k)}{\sum_{i=1}^n \mathbb{1}(y_i \geq t_k)}, \quad (40.4)$$

where $\{t_1, \dots, t_K\}$ is an ordered set of K unique failure times (y_i 's with $\delta_i = 1$) within the observed data, with t_1 as the minimum. Then, the estimated survival function $\hat{S}(t)$ can be calculated [4] by

$$\hat{S}(t) = e^{-\hat{\Lambda}(t)}. \quad (40.5)$$

Another commonly used estimator for the survival function is the Kaplan-Meier estimator [5], given by

$$\hat{S}(t) = \prod_{k:t_k \leq t} \left\{ 1 - \frac{\sum_{i=1}^n \mathbb{1}(y_i = t_k) \mathbb{1}(\delta_i = 1)}{\sum_{i=1}^n \mathbb{1}(y_i \geq t_k)} \right\}. \quad (40.6)$$

The two estimators are asymptotically equivalent [6] and are both widely used in survival analysis.

For simplicity, no covariate information is used in this section. In later sections, the covariate information will be used to build a random forest, and the following quantities can be used as an estimator at each terminal node and averaged over all trees of a random forest. For details on the estimation process, see Sect. 40.3.2.

40.1.2 Parametric and Semi-Parametric Models for Survival Data

For modeling with covariates, the Cox proportional hazards model [7] is the most popular choice. A Cox model assumes that the effect of covariates acts multiplicatively on the hazard function [8] such that

$$\lambda(t, \mathbf{x}) = \lambda_0(t) \exp(\mathbf{x}^T \boldsymbol{\beta}), \quad (40.7)$$

where $\lambda_0(t)$ is called a baseline hazard function. This assumption also implies that the log hazard ratio is a linear function of the covariates. Furthermore, using the relationships we introduced previously, the survival function can be defined as

$$S(t, \mathbf{x}) = \exp\{-\Lambda_0(t)\} \exp(\mathbf{x}^T \boldsymbol{\beta}), \quad (40.8)$$

where the baseline cumulative hazard $\Lambda_0(t) = \int_0^t \lambda_0(s) ds$. The Cox model is a semi-parametric model in the sense that the hazard (or cumulative hazard) function is modeled non-parametrically, while the effects of covariates are modeled parametrically. The nice property of a Cox model is that these parameters can be estimated by maximizing the partial log-likelihood function, which does not involve estimating the hazard function. To this end, the proportional hazard assumption is crucial for formulating the Cox. Cox model usually works well in practice. However, the proportional hazard assumption can also be restrictive since it implies that the fitted survival curves of the two subjects cannot cross each other. If the underlying true curves cross, a Cox model may fail to capture the effect of the covariates. For an example of proportional versus nonproportional hazards, see Fig. 40.1. The pair in Fig. 40.1a is Exponential(e^X), with $X = 1$ in orange and $X = 2$ in blue. The effect of X satisfies the proportional hazard assumption since the hazard function of an exponential distribution is its rate parameter. The pair in Fig. 40.1b is Beta($\alpha = 5, \beta = 10$) for $X = 0$ in blue and Exponential(3) for $X = 1$ in orange. Their survival functions cross at around 0.4. This intersection may cause trouble for the parameter estimate of X , which does not display a strongly monotone effect on the hazard.

There are many alternatives to the Cox model, including the accelerated failure time model [9], the proportional odds model [10], the frailty model [11], and many others. However, all of these models are composed of specific parametric components and underlying assumptions that could be violated, especially in a high-dimensional setting. In contrast, a random forest model provides a more flexible nonparametric framework for model estimation. We will focus on

introducing a breed of random forest models that can handle right-censored survival outcomes. We start with introducing tree-based models which serve as essential components of a random forests.

40.2 Tree-Based Models

A random forest is an ensemble of trees. Hence, we will first introduce a single tree model as well as its estimation and prediction procedures. A tree model recursively partitions a p -dimensional feature space \mathcal{X} (e.g., $[0, 1]^p$) into disjoint rectangular terminal nodes [12]. The model starts with a root node \mathcal{X} containing all observations and proceeds by constructing binary splits in the form of $\mathbb{1}(\mathbf{x}^{(j)} \leq c)$, where $\mathbf{x}^{(j)}$ denotes the j th variable. It splits the current node into two child nodes based on the output of this indicator function. Note that this methodology does not only split the feature space but also separates the observations into two subsets, one for each child node. The process is then recursively applied to each child node until reaching some stopping criteria, which produces a terminal node that contains a few observations. Following the notation in [6], we denote the collection of terminal nodes in a tree as $\mathcal{A} = \{A_u\}_{u \in \mathcal{U}}$, where \mathcal{U} is a collection of indices. These terminal nodes satisfy two properties: they jointly cover the whole feature space, i.e., $\mathcal{X} = \bigcup_{u \in \mathcal{U}} A_u$, and they do not overlap, i.e., $A_u \cap A_l = \emptyset$ for any $u \neq l$. Throughout the following, we use \mathcal{A} to represent a single tree, i.e., one possible partition of the feature space \mathcal{X} resulting from a binary tree.

As a demonstration, we refer to Fig. 40.2. This tree was built using the R packages `rpart` [13] and `partykit` [14].

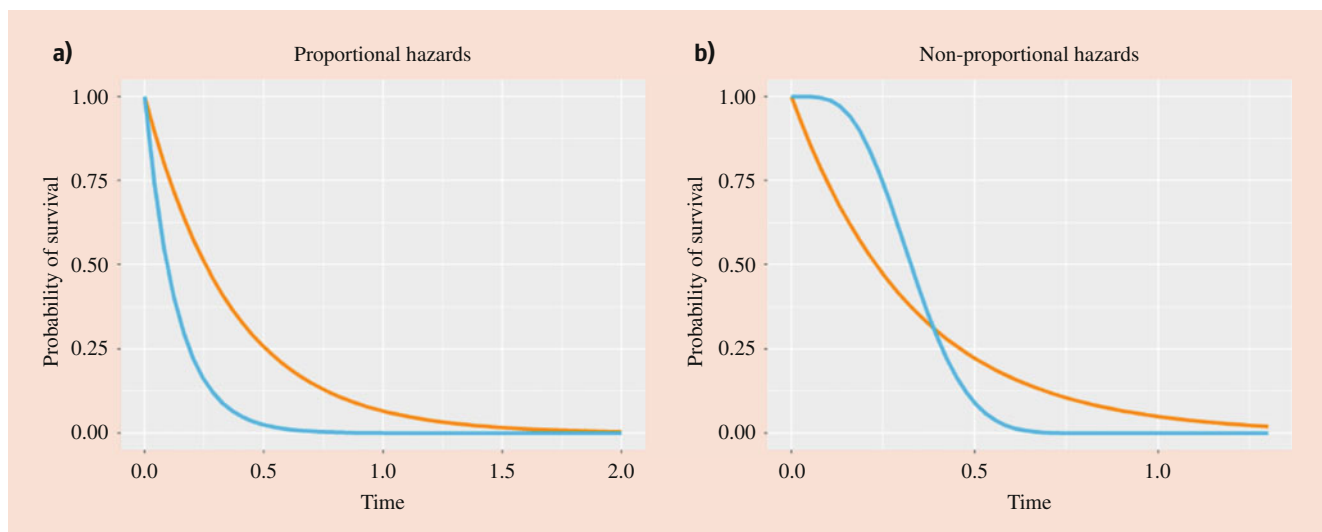


Fig. 40.1 Proportional versus nonproportional hazards. (a) is distributed as Exponential(e^X), where $X = 1$ is orange and $X = 2$ is blue. (b) is distributed as Exponential(3) when $X = 1$ is blue and Beta(5,10) when $X = 0$ is orange

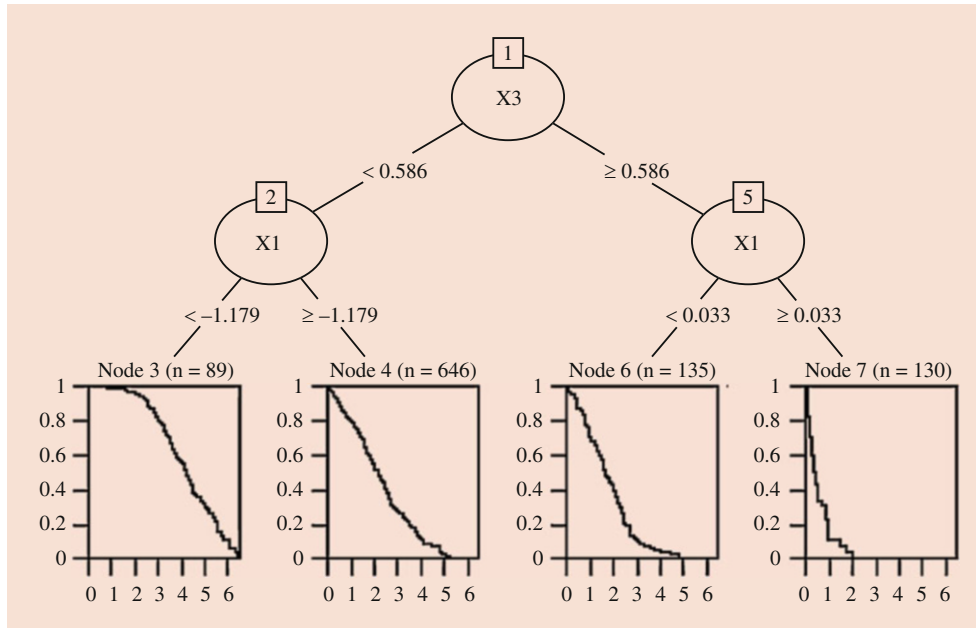


Fig. 40.2 A tree model, built from survival data with $p = 3$ covariates using `rpart` [13] and `partykit` [14]

In this example, $p = 3$ and $n = 1000$ (for more details on the generation of this data, see Sect. 40.6.2). The root node A , at the top of the figure and labeled 1, contains all $n = 1000$ observations. Then the tree has a binary split on $x^{(3)}$ at 0.586. If an observation i has a value of $x_i^{(3)} < 0.586$, it is sent to node A_L , labeled in Fig. 40.2 with a 2. Otherwise, it is sent to node A_R , labeled in this plot with a 5. Then the left and right nodes are each split one more time, creating four terminal nodes that contain 89, 646, 135, and 130 observations, respectively. The survival curves of each terminal node are estimated using the Kaplan-Meier estimator (40.6) and plotted as the terminal nodes. For example, the furthest left terminal node (labeled as *Node 3*) contains 89 observations where $x^{(3)} < 0.586$ and $x^{(1)} < -1.179$, and these limits do not overlap with any other node. It also has the highest estimated survival curve through the entire time domain, which implies that the observations that fall into that node are estimated to have the highest probability of survival. For this tree, $\mathcal{A} = \{A_3, A_4, A_6, A_7\}$, or $\mathcal{A} = \{A_u\}_{u \in \mathcal{U}}$ where $\mathcal{U} = \{3, 4, 6, 7\}$. Also note that, despite the fact $p = 3$, the tree never splits on $x^{(2)}$. Not all variables will necessarily be used for splitting. Hence, a tree model can potentially be a sparse model.

There are many types of trees. They mainly differ in terms of the criteria to find the splitting rule $\mathbb{1}(x^{(j)} \leq c)$. For example, [12] introduced classification and regression trees and [15] outlined survival trees. There is a rich literature of building survival trees, for example [16–18], and [19]. We refer to [20] for a comprehensive review. However, the tree building mechanisms are essentially the same for all these methods. We will introduce a couple of versions that can handle right-censored survival data.

40.2.1 Splitting Mechanism

At any internal node A of a tree \mathcal{A} , we search for a splitting rule $\mathbb{1}(x^{(j)} \leq c)$. This process leads to two child nodes: $A_L = A \cap \{x_i^{(j)} \leq c\}$, i.e., all the observations i from A that have a value of $x_i^{(j)} \leq c$ for a constant c , and $A_R = A \cap \{x_i^{(j)} > c\}$. The best split is the split that maximizes the difference between the observations in A_L and A_R . In the regression and classification settings, the variance reduction and Gini index [12] are commonly used to quantify this difference. In the survival context, any calculation of the differences between two potential child nodes has to account for censoring. In this section, we introduce the log-rank test statistic [1, 15] which is the default choice in many implementations, and an alternative choice, the supremum log-rank test [21], which is more sensitive to the distributional difference between two survival curves.

Log-Rank Test

The log-rank test is a two-sample test that compares the survival distributions of two groups when the observations are subject to censoring. It can be viewed as an analog of the two-sample t test in the regression setting. Here, the two potential child nodes are treated as the two groups in this test. To calculate the log-rank test statistic, we first define a few key quantities. Consider any node A , let $N(t)$ be the total number of at-risk observations at time t , i.e.,

$$N(t) = \sum_{i=1}^n \mathbb{1}(x_i \in A) \mathbb{1}(y_i \geq t), \quad (40.9)$$

and let $O(t)$ be the number of failure observations at time t , i.e.,

$$O(t) = \sum_{i=1}^n \mathbb{1}(x_i \in A) \mathbb{1}(y_i = t) \mathbb{1}(\delta_i = 1). \quad (40.10)$$

For both $N(t)$ and $O(t)$, we can correspondingly define the child node versions for the left and right nodes. Denote these quantities as $N_L(t)$, $N_R(t)$, $O_L(t)$, and $O_R(t)$, where L and R indicate the side. Furthermore, define

$$E_L(t) = O_L(t) \frac{N_L(t)}{N(t)} \quad (40.11)$$

to be the expected failure counts at time t for the left child node. This expectation is based on the intuition that the two groups have the same failure rate at time point t under the null hypothesis. Among the observed y_i values within an internal node A , we again use the notation $\{t_1, \dots, t_K\}$ as the ordered set of uniquely observed failure time points. We can then construct the log-rank test statistic [7] for a possible split as

$$\frac{\sum_{k=1}^K (O_L(t_k) - E_L(t_k))}{\sqrt{\sum_{k=1}^K E_L(t_k) \frac{N_R(t_k)}{N(t_k)} \frac{N(t_k) - O_L(t_k)}{N(t_k) - 1}}}. \quad (40.12)$$

It can be shown that this test statistic follows asymptotically a standard normal distribution, which leads to a valid p value. Essentially, the log-rank test statistic calculates the difference between the observed failure counts and the failure counts that would be expected if the groups came from the same distribution. In the case that the alternative hypothesis is proportional hazards (Fig. 40.1a), this statistic is asymptotically efficient for testing the difference between the two groups. However, this test may also have low power against some alternatives. For example, when the hazards are similar over most of the interval but diverge over a particular segment [21] or the expected values are the same (see Fig. 40.1b and Sect. 40.2.1), the log-rank test may fail. As an alternative, [21] developed a supremum version of the log-rank test statistic, hereafter referred to as the supremum log-rank test statistic.

Supremum Log-Rank Test

In the case of nonproportional hazards or when the distributions diverge but expected values are the same, the power of the supremum log-rank test is higher than that of the log-rank test. Using the same notation as the previous calculations, we can construct the supremum log-rank test [21, 22] statistic for a possible split as

$$\max_{j \in \{1, \dots, K\}} \frac{\left| \sum_{k=1}^j \frac{N_L(t_k) N_R(t_k)}{N(t_k)} \left(\frac{O_L(t_k)}{N_L(t_k)} - \frac{O_R(t_k)}{N_R(t_k)} \right) \right|}{\sqrt{\sum_{k=1}^K \frac{N_L(t_k) N_R(t_k)}{N(t_k)} \frac{O(t_k)}{N(t_k)} \left(1 - \frac{O(t_k) - 1}{N(t_k) - 1} \right)}}. \quad (40.13)$$

Instead of summing over every time point, it searches for a time point that displays the maximum divergence. Hence, the supremum log-rank test can be used to detect any distributional differences between two potential child nodes, rather than comparing the expected failures from the two groups.

Choosing the Best Split

For either the log-rank test or supremum log-rank test, a more significant p value indicates a better split. We exhaust all possible indices $j \in \{1, \dots, p\}$ of the splitting variables and cutting point c from all unique $\mathbf{x}^{(j)}$ values within node A to define the two groups. The splits that leads to the most significant test is then used. However, this exhaustive search may not be required when fitting a random forest model. Section 40.3.3 introduces the details of these modifications. As proposed by Ishwaran et al. [23], the splitting will stop when there are too few failure observations in a node.

A Toy Example

To demonstrate the two splitting rules, we will use a set of samples from the nonproportional hazard distribution setting shown in Fig. 40.1b. We randomly sample 60 failure times T_{is} , with 30 of them from Exponential(3) and another 30 from Beta(5,10). We use a corresponding covariate value $x_i^{(1)} = 0$ or 1 to indicate their group. For all observations, the censoring times C are generated independently from a Uniform(0,1) distribution to calculate the observed time y and censoring indicator δ . Therefore, the maximum observed time is less than or equal to 1, as the censoring time has an upper bound. Additionally, we generated another group indicator $x_i^{(2)}$ with $P(x_i^{(2)} = 0) = P(x_i^{(2)} = 1) = 0.5$ independently from all other quantities. This covariate mimics the situation that we might be facing with other noise variables that are not associated with survival time. For more details on this simulation, see the sample code provided in the supplementary material (Sect. 40.8).

The log-rank test statistic using $x^{(1)}$ as the group indicator is 1.57, with a p -value of 0.2. When we use $x^{(2)}$ as the group indicator, we get a log-rank test statistic of 3.64, with a p -value of 0.06. For a split on this node using these results, we would choose to split on $x^{(2)}$ with a cutting point between 0 and 1. However, take a look at Fig. 40.3 again. The true distributions of both groups have an expected survival time equal to 1/3, but these two groups do not have the same true survival distribution. There is about an equal area where the orange line is greater than the blue line as the opposite. Therefore, the log-rank test has trouble detecting the difference between these particular two groups. On the other hand, the supremum log-rank test for $x^{(1)}$ yields a p -value of 0.05 compared to $x^{(2)}$ with a p -value of 0.11. As the supremum log-rank test searches over the domain for the section of greatest divergence, it selects the signal variable first.

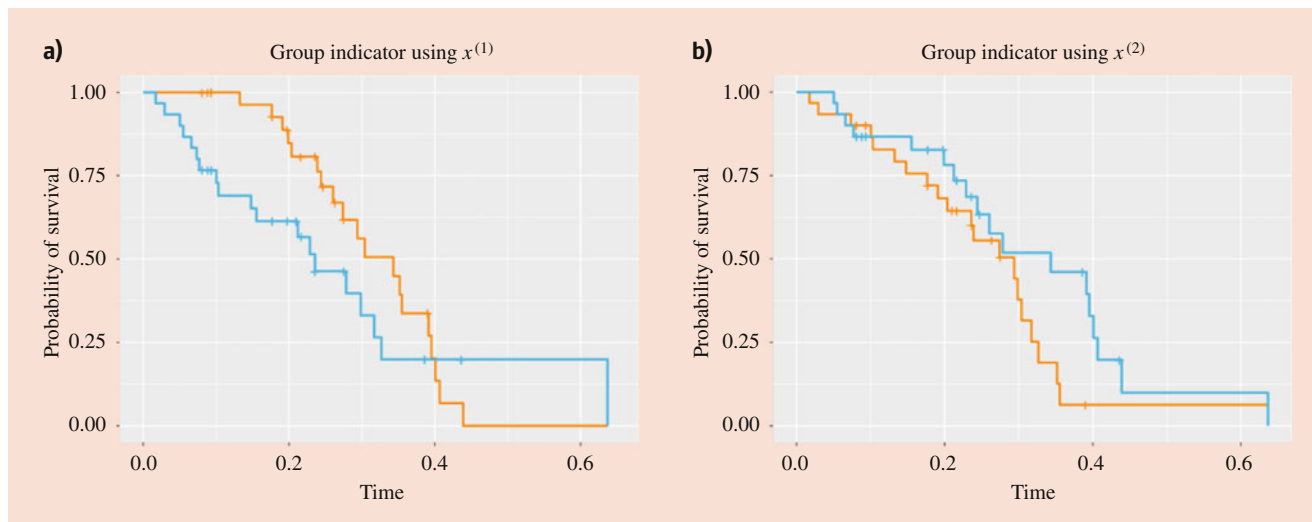


Fig. 40.3 (a): A sample of 30 observations from each of the distributions referenced in 40.1b. $x^{(1)} = 0$ is from Exponential(3) (orange), $x^{(1)} = 1$ is from Beta(5,10) (blue). (b): the same sample as in (a),

but $x^{(2)}$ is randomly assigned as 0 (orange) or 1 (blue). “+” indicates censored observations. The survival curves are estimated using the Kaplan-Meier estimator (40.6)

Other Splitting Rules

There are many variations on survival trees and survival splitting rules. Besides the literature above, [24] uses the imputation of censored data to improve prediction, and [6,25] develop splitting rules that correct the bias caused by the censoring distribution. For more splitting rules that can account for bias, see Sect. 40.6. After splitting is complete, one possible next step is to prune the tree: starting from the fully fitted tree, remove unnecessary branches to prevent overfitting. For example, the cost-complexity pruning is used in [12] for the regression setting. For general concepts of overfitting in statistical learning algorithms, we refer to [26].

40.2.2 Tree Prediction

Using a partition of the feature space, we can produce predictions for any target point $\mathbf{x}_0 \in \mathcal{X}$. For survival analysis, the prediction for \mathbf{x}_0 is simply the survival function estimator as defined in (40.4) or (40.6) applied to the data from the terminal node that contains \mathbf{x}_0 . For example, the Nelson-Aalen estimator for the target point is given by

$$\widehat{\Lambda}(t, \mathbf{x}_0) = \sum_{k:t_k \leq t} \frac{\sum_{i=1}^n \mathbb{1}(\delta_i = 1) \mathbb{1}(y_i = t_k) \sum_u \mathbb{1}(\mathbf{x}_0 \in A_u) \mathbb{1}(\mathbf{x}_i \in A_u)}{\sum_{i=1}^n \mathbb{1}(y_i \geq t_k) \sum_u \mathbb{1}(\mathbf{x}_0 \in A_u) \mathbb{1}(\mathbf{x}_i \in A_u)}. \quad (40.14)$$

This formula resembles a form of the kernel estimator [27, 28] (see Sect. 40.7 for more details), making tree-based methods nonparametric by nature. From this point forward,

we will use the shortened notation, which indicates whether two points \mathbf{x}_i and \mathbf{x}_j ever fall into the same terminal node in a given tree:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{u \in \mathcal{U}} \mathbb{1}(\mathbf{x}_i \in A_u) \mathbb{1}(\mathbf{x}_j \in A_u). \quad (40.15)$$

Then the Nelson-Aalen formula for a terminal node becomes

$$\widehat{\Lambda}(t, \mathbf{x}_0) = \sum_{k:t_k \leq t} \frac{\sum_{i=1}^n \mathbb{1}(\delta_i = 1) \mathbb{1}(y_i = t_k) K(\mathbf{x}_0, \mathbf{x}_i)}{\sum_{i=1}^n \mathbb{1}(y_i \geq t_k) K(\mathbf{x}_0, \mathbf{x}_i)}. \quad (40.16)$$

However, a tree model and its induced kernel function are not smooth because there are abrupt jumps in the prediction once the target point moves from one terminal node to another. To introduce a smoother and more stable model, we can build many trees with variations and average them.

40.3 Random Forests for Right-Censored Survival Data

40.3.1 An Ensemble of Trees with Randomness

Random forest proposed by Breiman [29] is a prediction model that consists of a collection of trees, with bootstrap samples [30] and random feature selections [31, 32] incorporated into each tree. The tree construction process stops when the node sample size is sufficiently small, and the algorithm does not perform pruning. Each tree produces a prediction model for the outcome variable, and the forest generates

a prediction from the average of the tree predictions [29] in the regression setting, while majority voting is used for classification. Random forests have been adapted to many settings besides regression and classification, such as survival analysis [23, 24, 33], quantile regression [34], unsupervised learning [35], etc.

Random forests [29] utilize two main mechanics to control for over-fitting and to reduce correlations among trees: bagging and random feature selection. To build each tree, the bagging (bootstrap aggregating) procedure takes bootstrap samples from the original data set [30]. Then trees are grown using each bootstrap sample. These trees are then averaged to produce a forest prediction. Note that other implementations also consider sampling without replacement [36]. These choices may affect the calculation of other quantities, such as the variable importance measure that will be introduced later in Sect. 40.4.

The second main mechanic for controlling over-fitting is the random feature selection. This procedure takes place within the tree-building process. When trying to find a split at any internal node, the algorithm considers only a set of randomly selected `mtree` variables without replacement. The splitting variables will be searched only within this set at this internal node. When proceeding to the next node, this set will be regenerated. This method is motivated by works such as [31] and [32] which lead to less correlated trees. By the corresponding statistical theory, averaging weakly correlated estimators (trees) results in a smaller variation of the averaged estimator (random forest). Hence, the prediction accuracy of a random forest may be significantly better than a simple tree version.

We note that there are other possible random mechanisms such as the extremely randomized trees proposed by Geurts et al. [37] that considers choosing random cut points when splitting an internal node. In this case, one or several random cut points are generated for each variable. The splitting test statistic is calculated for each of them, and the best split among them is selected. This mechanism further reduces the correlation among trees and has a significant computational advantage due to less number of evaluations of the test statistics.

40.3.2 Ensemble Prediction

Random forest prediction is, in most cases, an averaged tree predictor. For example, the prediction for a target point based on a single tree is from one of the terminal nodes in a tree $\mathcal{A} = \{A_u\}_{u \in \mathcal{U}}$. For a random forest, we have a collection of B such trees, which can be denoted as $\{\mathcal{A}_b\}_{b=1}^B = \{\{A_u^b\}_{u \in \mathcal{U}^b}\}_{b=1}^B$. Based on Eq. (40.14), we obtain a set of predictions $\hat{\Lambda}_b(t, \mathbf{x}_0)$, $b = 1, \dots, B$, calculated using

$$\hat{\Lambda}_b(t, \mathbf{x}_0) = \sum_{k:t_k \leq t} \frac{\sum_{i=1}^{n_b} \mathbb{1}(\delta_i^b = 1) \mathbb{1}(y_i^b = t_k) K_b(\mathbf{x}_i, \mathbf{x}_0)}{\sum_{i=1}^{n_b} \mathbb{1}(y_i^b \geq t_k) K_b(\mathbf{x}_i, \mathbf{x}_0)}, \quad (40.17)$$

where $\{\mathbf{x}_i^b, y_i^b, \delta_i^b\}_{i=1}^{n_b}$ is the set of bootstrap/resampling observations used in the b th tree and $K_b(\mathbf{x}_i, \mathbf{x}_0) = \sum_{u \in \mathcal{U}^b} \mathbb{1}(\mathbf{x}_i \in A_u^b) \mathbb{1}(\mathbf{x}_0 \in A_u^b)$ is the corresponding kernel (see Eq. (40.15)) that indicates whether \mathbf{x}_i and \mathbf{x}_j fall in the same terminal node in tree b . Then the forest survival function estimator is simply aggregating these tree estimators with

$$\hat{\Lambda}(t, \mathbf{x}_0) = \frac{1}{B} \sum_{b=1}^B \hat{\Lambda}_b(t, \mathbf{x}_0), \quad (40.18)$$

while the survival function can be calculated using

$$\hat{S}(t, \mathbf{x}_0) = e^{-\hat{\Lambda}(t, \mathbf{x}_0)}. \quad (40.19)$$

We may also consider alternative definitions of the survival function estimation. For example, [24] considered directly averaging the Kaplan-Meier estimators [5] (40.6) from each tree. Another possibility is to view the entire forest as a single kernel, stacked from each individual tree. Then the kernel version of the cumulative hazard function can be defined as

$$\hat{\Lambda}(t, \mathbf{x}_0) = \sum_{k:t_k \leq t} \frac{\sum_b \sum_{i=1}^{n_b} \mathbb{1}(\delta_i^b = 1) \mathbb{1}(y_i^b = t_k) K_b(\mathbf{x}_i, \mathbf{x}_0)}{\sum_b \sum_{i=1}^{n_b} \mathbb{1}(y_i^b \geq t_k) K_b(\mathbf{x}_i, \mathbf{x}_0)}. \quad (40.20)$$

Similar approaches of the kernel version of random forests have been considered in the regression [38] and generalized versions [39]. However, it has yet been reported if any of the aforementioned averaging method lead to superior performance.

40.3.3 Tuning Parameters

Tuning parameter selection is probably the most important part in the application of random forests. There are many R packages such as `randomForest` [40], `randomForestSRC` [41], and `ranger` [42] that offers a variety of tuning parameter. We will use a newly developed R package `RLT` [43] as demonstration. The `RLT` package offers both standard tuning parameters and addition features that improve the model fitting process and computational performance. It is available on GitHub. The major tuning parameters involved in a random forest aim at controlling the randomness across different trees and the splitting mechanism.

- Sampling size (`resample.prob`): The proportion of the total sample used for building each tree. In RLT, `resample.prob = 1` means each tree is fitted using n observations.
- Sampling type (`resample.replace`): Whether the resampling of observations for a tree is done with or without replacement. In RLT, `resample.replace=TRUE` indicates sampling with replacement.
- Number of trees (`ntrees`): The default value for `ntrees` is 500. This value should be set as a relatively large number so that the averaged model is stable.
- Number of random features sampled (`mtry`): This implements the random feature selection at each internal node and only compares these selected features to search for the best splitting rule. Common settings are `mtry = \sqrt{p}` , `$p/3$` , and possibly `p` , which exhausts all variables.
- Splitting score (`split.rule` under `param.control` settings): The type of measure used to pick the best split. The options for survival analysis are "logrank" (40.12) and "suplogrank" (40.13).
- Type of search for the cutting point (`split.gen`): The method used to generate potential cutting points for a splitting rule. The common choices are `split.gen = "best"` or "random". "best" is the mechanism described Sect. 40.2.1: it considers all possible splits. "random" generates a set of random cutting point and selects the best among them. This setting mimics the mechanism proposed in [37].
- Number of random splits (`nsplit`): if `split.gen = "random"`, this parameter controls the number of randomly generated cutting points. This number is usually small to achieve computational efficiency.
- Terminal node size (`nmin`): Splitting will not occur on a node with less than $2 \times nmin$ observations. Under random splitting, there may not be at minimum `nmin` observations at a terminal node. For survival analysis, it can be beneficial to require $2 \times nmin$ number of observed failure events to perform a split.
- Child node size control (`alpha` under `param.control` settings): At an internal node, if the sample size is n , `alpha` ensures each child node contains at least `alpha \times n` observations. This parameter is not effective when the `split.gen = "random"`. Setting a nonzero `alpha` value can ensure additional theoretical properties especially when `nmin` is small.
- Variable importance control (`importance`): Whether variable importance (see Sect. 40.4) should be calculated. In RLT, `importance=FALSE` will not calculate variable importance, which will make the algorithm faster if this measure is not desired for the analysis.

Finally, we provide a high-level algorithm for general random forest models in Algorithm 1. Note that for different

types of random forest models, the main differences are the splitting rule and terminal node estimation.

Algorithm 1: General algorithm for random forest models

Input: Training data set \mathcal{D}_n and parameters from Sect. 40.3.3.
Let $B = n_{trees}$.

```

1 for  $b = 1$  to  $B$  do
2   Initiate  $A = \mathcal{X}$ , a bootstrap sample  $\mathcal{D}_n^b$  of size
   resample.prob  $\times$  n from  $\mathcal{D}_n$ , and the set of terminal
   nodes  $\mathcal{A}_b = \emptyset$ ;
3   At a node  $A$ , if  $\sum_{x_i \in \mathcal{D}_n^b} \mathbb{1}(x_i \in A) < 2 \times nmin$ , proceed
   to Line 5. Otherwise, construct a splitting rule
   (Sect. 40.2.1) such that  $A = A_L \cup A_R$ , where  $A_L \cap A_R = \emptyset$ 
   and  $\min(\sum_{x_i \in A} \mathbb{1}(x_i \in A_L), \sum_{x_i \in A} \mathbb{1}(x_i \in A_R)) >$ 
   alpha  $\times$   $\sum_{x_i \in \mathcal{D}_n^b} \mathbb{1}(x_i \in A)$ ;
4   Send the two child nodes  $A_L$  and  $A_R$  to Line 3 separately;
5   Conclude the current node  $A$  as a terminal node and update
    $\mathcal{A}_b = \mathcal{A}_b \cup A$ .
6 end
7 return  $\{\mathcal{A}_b\}_{b=1}^B$ .
```

40.3.4 Evaluation Criteria for Survival Models

For regression, the commonly used error is the mean squared error. Classification models often use the misclassification rate. When calculating the error for a survival model, we have to compare a fitted function (hazard or survival function) with the observed survival outcomes. There are two popular survival error metrics used in practice: the C-index error [44] and the Brier Score error [45]. Other possible choices, such as the Cox-Snell residual and the martingale residual, are more commonly used for diagnosis in semi-parametric models.

The C-index is essentially a classification measure adjusted for censoring. It measures the probability that, for two randomly drawn subjects, the subject with the lower survival time is classified correctly as having a worse prediction (higher hazard function or lower survival function) than the other subject. However, since these functions are varying according to t , we need a scalar summary of the prediction to indicator whether a prediction is better or worse. As proposed in [23], this prediction is defined as

$$v_i = \sum_{k=1}^K \widehat{\Lambda}(t_k, \mathbf{x}_i), \quad (40.21)$$

for each subject i . Hence, $v_i < v_j$ indicates that subject j has a worse predicted outcome than subject i . Then, we calculate the ratio of corrected classified pairs over permissible pairs, where the number of permissible pairs is defined as

$$P = \sum_{\substack{i < j: y_i \neq y_j, \\ \delta_i = 1 \text{ if } y_i < y_j, \\ \text{or } \delta_j = 1 \text{ if } y_j < y_i}} 1 + \sum_{\substack{i < j: y_i = y_j \\ \text{and } \delta_i + \delta_j > 0}} 1,$$

and the number of correctly classified pairs is

$$\begin{aligned}
 C = & \sum_{\substack{i < j: y_i < y_j \\ \delta_i = 1}} \left\{ \mathbb{1}(v_i > v_j) + 0.5 \cdot \mathbb{1}(v_i = v_j) \right\} \\
 & + \sum_{\substack{i < j: y_j < y_i \\ \delta_j = 1}} \left\{ \mathbb{1}(v_i < v_j) + 0.5 \cdot \mathbb{1}(v_i = v_j) \right\} \\
 & + \sum_{\substack{i < j: y_j = y_i \\ \delta_i = 1, \delta_j = 1}} \left\{ \mathbb{1}(v_j = v_i) + 0.5 \cdot \mathbb{1}(v_j \neq v_i) \right\} \\
 & + \sum_{\substack{i < j: y_j = y_i \\ \delta_i + \delta_j = 1}} 0.5 \cdot \left\{ 1 + \mathbb{1}(v_j > v_i) \mathbb{1}(\delta_j = 1) + \mathbb{1}(v_i > v_j) \mathbb{1}(\delta_i = 1) \right\}.
 \end{aligned}
 \tag{40.22}$$

Then the C-index for a prediction function \hat{f} is simply

$$\text{C-index} = \frac{C}{P}, \tag{40.23}$$

with larger values indicating better model fits. The C-index error is then $1 - C/P$.

Another choice is the Brier score, which was first introduced by Brier [46] as a classification measure and eventually adjusted for survival [45]. Unlike the C-index, the Brier score requires the conditional censoring distribution $G(t, \mathbf{x}) = P(C > t | \mathbf{X} = \mathbf{x})$ to be specified. Since the true censoring distribution is unknown, the performances of a Brier score can be affected by the choice of the censoring model to estimate this distribution. In practice, this is often simplified to a marginal censoring distribution $G(t) = P(C > t)$, which can be estimated through a Kaplan-Meier estimator (40.6) using the data with $1 - \delta_i$ as the censoring indicator. Alternatively, one can estimate the conditional function $G(t, \mathbf{x})$ using a Cox proportional hazard model with the same strategy of reversing the censoring indicator. Using an estimator $\hat{G}(t, \mathbf{x})$, the Brier Score at time t can be calculated as

$$\begin{aligned}
 \text{BS}(t) &= \frac{1}{n} \sum_{i=1}^n \left(\frac{\mathbb{1}(y_i \leq t) \mathbb{1}(\delta_i = 1) \hat{S}(t, \mathbf{x}_i)^2}{\hat{G}(y_i^-, \mathbf{x}_i)} + \frac{\mathbb{1}(y_i > t) (1 - \hat{S}(t, \mathbf{x}_i))^2}{\hat{G}(t, \mathbf{x}_i)} \right),
 \end{aligned}
 \tag{40.24}$$

where y^- is the left limit to incorporate the fact that $\hat{G}(t, \mathbf{x})$ is often times not smooth. Note that the Brier Score is only defined on a single time point, we often use the integrated Brier Score to summarize the errors. This is defined as

$$\text{IBS}(\hat{f}) = \int_{t=0}^{\tau} \text{BS}(t) dt, \tag{40.25}$$

where τ is a pre-chosen maximal time point, e.g., the maximum observed failure time. The integral has to be calculated as a summation over all observed time points in practice.

It should be noted that if the observed data are extremely noisy, empirical evaluations of both C-index and Brier score could be insensitive to the performance.

40.3.5 Out-of-Bag Error

When using sampling with replacement (bootstrap) in random forests, a sample size of n consists of approximately $0.632 \times n$ unique samples. This is based on the fact that each observation has a $1 - 1/n$ chance not to be selected if we randomly pick one observation and the probability of being in the testing data is then $(1 - 1/n)^n$, which converges to e^{-1} . This means that each tree excludes about one-third of the data. These excluded observations are called the out-of-bag samples (corresponding to the particular tree). It also allows calculation of prediction error by treating them as an independent testing set against the corresponding tree. This testing set is particularly useful when evaluating the model performance and selecting variables. The resulting test error is referred to as the out-of-bag error for the corresponding tree. The mechanism of out-of-bag error is essentially a type of cross-validation. However, as an ensemble method, random forests have their unique way of aggregating these out-of-bag errors over the entire forest. Hence it does not require explicitly splitting the data into training and testing samples.

For each of the training observations, we may look at all the trees that do not use that observation for building the tree, i.e., the observation is in the out-of-bag sample of these trees. As long as `resample.replace=TRUE` or `resample.prob<1`, there will be an out-of-bag sample, and the out-of-bag error can be calculated. Denote the collection of such out-of-bag samples as $\{\mathcal{L}_b^{\text{oob}}\}_{b=1}^B$. The out-of-bag prediction for \mathbf{x}_i is the average prediction over the trees that exclude \mathbf{x}_i . In the survival case, (40.18) would be modified to

$$\hat{\Lambda}_{\text{oob}}(t, \mathbf{x}_i) = \frac{1}{\sum_{b=1}^B I(\mathbf{x}_i \in \mathcal{L}_b^{\text{oob}})} \sum_{b=1}^B I(\mathbf{x}_i \in \mathcal{L}_b^{\text{oob}}) \hat{\Lambda}_b(t, \mathbf{x}_i). \tag{40.26}$$

This prediction can then be compared with the observed outcome value $\{y_i, \delta_i\}$ to calculate a prediction error, such as the C-index or Brier score.

40.4 Variable Selection via Variable Importance in High-Dimensional Settings

High-dimensional data are commonly observed in biomedical studies. Gene expression data sets often consist of 30K~35K variables. Variable selection tools are highly

desirable for handling large amounts of variables. Random forests provide a built-in mechanism, the variable importance measure, for estimating and ranking the contribution of each variable. This measure is again a by-product of the bootstrapping and out-of-bag prediction errors. The basic procedure works as follows:

- For the b th tree, calculate the prediction error of the out-of-bag samples $\mathcal{L}_b^{\text{oob}}$ using the fitted tree. Denote the out-of-bag error as Err_b for $b = 1, \dots, B$.
- For each tree, take the out-of-bag samples and randomly shuffle the values of a variable j within these samples. Using this newly created out-of-bag data, evaluate the prediction error. Denote the errors as $\text{Err}_b^{(j)}$, for $b = 1, \dots, B$ and $j = 1, \dots, p$.
- Calculate the j th variable importance as

$$\text{VI}(j) = \frac{1}{B} \sum_{b=1}^B \frac{\text{Err}_b^{(j)}}{\text{Err}_b} - 1.$$

Alternatively, one can also use $B^{-1} \sum_{b=1}^B (\text{Err}_b^{(j)} - \text{Err}_b)$. In practice, we may use any of the previously introduced error metrics for Err_b . The logic behind the variable importance measure is that after randomly shuffling a variable j within the out-of-bag data, its association with the outcome is destroyed. Hence, if this variable is initially informative for predicting the survival time, the prediction error $\text{Err}_b^{(j)}$ should be much larger than Err_b calculated from original data due to this loss of information. Hence, by taking the averaged ratio (or difference) of the two, and subtracting one, the variable importance measure should be positive. On the other hand, if a variable is non-informative for predicting the outcome, we would expect its variable importance to be approximately zero. Theoretical analysis of the variable importance under regression settings has been done by Zhu et al. [43], which essentially explains these properties. However, note that the variable importance relies on the behavior of the fitted tree models. Therefore, random forest with different turnings may lead to different rankings of the variable importance. As a consequence, the variable importance from a random forest should be interpreted as the importance reflected from the model itself, rather than the importance of the underlying truth. Of course when random forest can consistently estimate the underlying model [47], the two concepts are asymptotically the same. These subtle differences are often ignored in practice.

40.4.1 Other Methods for Calculating Variable Importance

The process mentioned above (hereafter referred to as “permutation variable importance”) is not the only way to

determine which variables have more predictive power. The most naive approach is simple: count the number of times each variable is used as a splitting variable within the forest and rank accordingly [48]. However, the naive method is strongly biased toward selecting continuous variables, as these variables have more places to split.

One potential issue with permutation variable importance arises when the variables are correlated [49]. For example, let $\mathbf{x}^{(j)}$ and $\mathbf{x}^{(l)}$ be two correlated variables. When $\mathbf{x}^{(j)}$ is permuted, $\mathbf{x}^{(l)}$ will still have its original values. Therefore, $\mathbf{x}^{(l)}$ may help correctly classify the observations even without $\mathbf{x}^{(j)}$. In that case, the importance of correlated predictor variables may be suppressed. It should be noted that, from a statistical point of view, these issues are mainly caused by an identifiability issue, in which the focus is variable selection. However, from a prediction point of view, one may be interested in both variables regardless of which is the truth. Hence, again, a variable importance measure may be better interpreted as the importance presented in the current fitted model as we explained previously.

Nonetheless, some existing methods can be used to deal with correlated variables, such as [49] who developed conditional variable importance. Their process is a variation on permutation variable importance that includes conditioning on a second variable. For a variable j , determine all other variables $Z = \{j^{(1)}, j^{(2)}, \dots, j^{(q)}\}$ that are correlated with j . It is suggested in [49] to use only the variables whose correlation with variable j exceeds a certain threshold. Then we calculate conditional variable importance as follows:

- For the b th tree, calculate the prediction error of the out-of-bag samples $\mathcal{L}_b^{\text{oob}}$ using the fitted tree. Denote the out-of-bag error as Err_b for $b = 1, \dots, B$.
- For the b th tree, for each variable in the index set Z , extract the cutting points used to split it and construct a grid by bisecting the sample space using all variables and their corresponding cutting points.
- Randomly shuffle the values of a variable j within each section of the grid. Using this newly shuffled data, evaluate the out-of-bag prediction error. Denote the errors as $\text{Err}_b^{(j|Z)}$, for $b = 1, \dots, B$.
- Calculate the conditional variable importance of the j th variable as

$$\text{VI}(j|Z) = \frac{1}{B} \sum_{b=1}^B \frac{\text{Err}_b^{(j|Z)}}{\text{Err}_b} - 1$$

One obvious problem is that variables j and Z may not occur in the same tree. However, [49] mentioned that with enough trees, the different combinations should appear often enough for an accurate estimate. While [49] developed the method for regression forests, the same process can be adapted to survival forests.

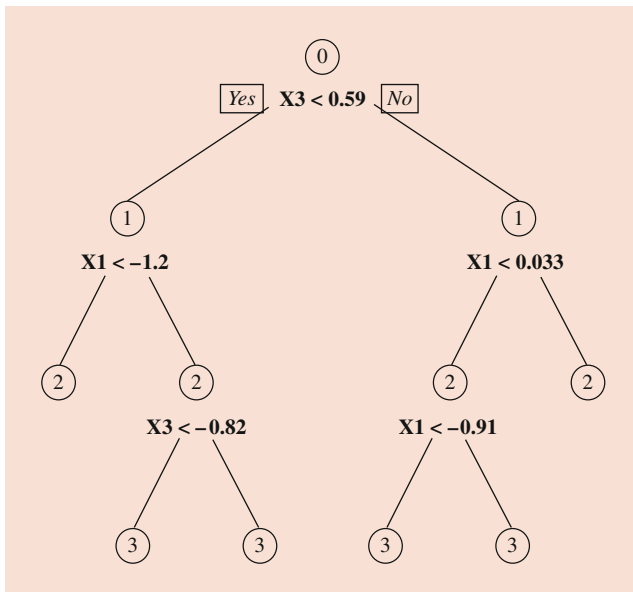


Fig. 40.4 A survival tree with $p = 3$ covariates. Nodes are labeled with depth measure

Another method for variable selection relies on using maximal sub-trees [50] for a variable $x^{(j)}$, defined as “the largest sub-tree (a connected tree within a tree) whose root node is split using” the variable of interest. The associated measure of the variable $x^{(j)}$ ’s strength is the distance from the root of the tree to the maximal sub-tree of $x^{(j)}$. As a smaller distance indicates that $x^{(j)}$ is closer to the root, a smaller value indicates a variable with higher predictive power. For example, consider the tree in Fig. 40.4 from the model defined in Sect. 40.6.2. We generated a plot using `rpart` [13] and `rpart.plot` [51]. There are three sub-trees for $x^{(1)}$: one starting at the split $x^{(1)} < -1.2$, a second at $x^{(1)} < 0.033$, and lastly, one at $x^{(1)} < -0.91$. These three trees have depth 1, 1, and 2, respectively, from the root node. Therefore, the minimal depth of $x^{(1)}$ is 1. $x^{(3)}$ has two sub-trees: one starting at the root node and the other at the split $x^{(3)} < -0.82$. These trees have depth 0 and 2, respectively, for a minimum depth of 0. As $x^{(2)}$ is not used in the tree, its minimum depth is infinity. The advantage of this minimal depth importance is that it does not depend on the particular type of outcome. Hence, it does not rely on the accuracy of the error measurement. Again, all the variable importance measures introduced here rely on the accuracy of the fitted tree model and should be viewed as a way of interpreting the black-box algorithm.

40.5 Example: Breast Cancer Data Analysis

We use the data collected by Wang et al. [52] for analyzing breast cancer relapse time as an illustration of random survival forest. The data is available on the

Gene Expression Omnibus [53] under the ascension number GSE2034 (<https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE2034>). The goal of this analysis is to provide a better way to assess the risk of relapse for patients with lymph-node-negative breast cancer [52]. The code to pull the data, the data itself, and the full code for our analysis are available on Sarah E. Formentini’s GitHub repository (<https://github.com/sarahef2/springer-surv>). We used the RLT package, which is available on GitHub (<https://github.com/teazrq/RLT>) to fit the survival random forest model. In the analysis from [52], the authors used the estrogen receptor (ER) status (+/–) as a stratification factor for analysis. A univariate Cox proportional hazards model was used to identify genetic covariates associated with relapse-free survival for ER+ and ER– individually. The authors further identified 60 genes related to ER+ patients and 16 genes related to ER– patients. Together, the authors refer to this group of 76 genes as a “gene signature.” They show this gene signature to be powerful for predicting high risks of breast cancer relapse [52].

40.5.1 Data Processing and Description

The data has 22,283 genes, the ER status, and the time until relapse for the 286 patients. For this example, we did not use the full 22,283 genetic variables. As [52] defined a 76 variable gene signature, we include these variables in our analysis. We additionally selected the top 500 variables using marginal screening via a Cox proportional hazards model, similar to the initial analysis step in [52], but not stratified by ER status. This selection leads to a total of 552 unique genetic variables in our analysis. To mirror the study in [52], we also included ER status as a covariate. None of the other clinical variables such as age are available online with the rest of the data, so they are not included.

A plot of the marginal survival function stratified by the ER status of patients is given in Fig. 40.5. The patients’ censoring times are marked with “+” on the survival functions. There are only 107 failures in the data set, which gives a total failure rate of 37%. The median observed time until relapse is 28 months, and the median censoring time is 104 months.

40.5.2 Fitting Survival Random Forest

To fit the model, we first consider a set of tuning parameters to select the best tuning. We consider `mtry = \sqrt{p} , $p/3$, and p , $nmin = 1, \log(n)$, and $n^{1/3}$, split.rule=logrank and suplogrank, and split.gen=best and random. For each tuning, we use 20,000 trees, sampling with replacement, and a bootstrap sample size of n . The rest of the tuning parameters were left as their package defaults.`

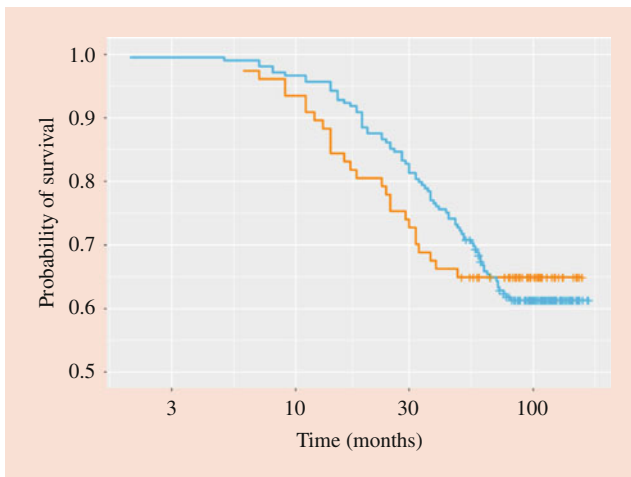


Fig. 40.5 Marginal survival function until relapse, stratified by ER status. The *orange line* is ER+; the *blue line* is ER-. “+” indicates censored observations

Using the out-of-bag C-index error as the selection criteria, the best combination of parameters is `mtry = $p/3 \approx 184$` , `nmin = 1`, `split.rule=logrank`, and `random split`, with an out-of-bag C-index error of about 0.262. The worst combination, `mtry = p` , `nmin = 1`, `split.rule=logrank`, and `best split`, has an out-of-bag C-index error of 0.296, which is 13% larger. In this case, the worst combination of parameters was the set that tried every single variable and every cutting point on that variable in every node, as well as only requiring a single observation in each terminal node. That model likely overfit to the noise in the fitting data, making it a poor predictor to new data. Additionally, `split.rule=suplogrank` performed rather similarly to `split.rule=logrank`. Alternatively, we could have chosen the best set of parameters using k -fold cross-validation. However, that would have required running the algorithm k times for each combination of parameters, which would be much more computationally expensive.

Using the best tuning parameter settings, we ran the final model and calculated variable importance. Of the top 40 variables by importance (Fig. 40.6), 4 are among the 76 gene signature provided by Wang et al. [52]. The other 36 are from marginal screening. Finally, we plot the fitted survival functions of four subjects. These four subjects were selected based on ER status (“+” and “-”) and the variable with the highest importance, the genetic variable 219478_at (high or low). The chosen subjects have the maximum or minimum value of 219478_at within the ER+ and ER- groups. Their out-of-bag predicted survival curves are plotted in Fig. 40.7. The ER+ subject with a high level of 219478_at has a shorter predicted survival time than the ER+ subject with a low 219478_at value, and a similar effect is observed with the ER- subjects. A large divergence in the most important variable, 219478_at, led to very different predictions.

40.6 Bias and Corrections of Survival Forests

Ideally, we would observe the time until the event in every case. However, due to censoring, we only have incomplete data about the time until the event. This is the central concern of survival analysis. However, parametric and semi-parametric models for handling censored data could be potentially biased if the model is mis-specified. For example, the Cox proportional hazard model is built to handle right censored data. However, when there is no censoring, it does not reduce to a standard linear regression model [54]. The resulting bias is especially pronounced when the hazards are not proportional. Some methods, such as the accelerated failure time model [55], do reduce to an ordinary linear regression model when there is no censoring. However, the problem of bias is not confined solely to linear models.

40.6.1 Biasedness of Survival Forests

In many cases, especially in the case of genetics data, many of the variables may be weakly (or even strongly) correlated. In particular, some variables that affect failure times (“failure variables”) may be correlated with variables that affect censoring times (“censoring variables”). Ideally, we would split primarily on failure variables rather than censoring variables. However, as long as there is a weak dependence, splitting on censoring variables may be unavoidable using the previously mentioned methods [6]. This phenomenon may cause biased tree structure, as well as biased estimations.

Random forest models also suffer biasedness although they are nonparametric in nature. This is mainly caused by the single variable splitting rule. A splitting rule at any internal node essentially compares the survival curves computed from two potential child nodes. Most existing analyses of the Kaplan-Meier estimator assume that the observations are i.i.d. [56, 57] or at least one set of the failure times or censoring times are i.i.d. [58]. However, these assumptions are almost always not true for tree-based methods because both T_i and C_i typically depend on some shared covariates. Thus, by leveraging the standard Kaplan-Meier estimator or Nelson-Aalen estimator, the child node estimations of the survival functions are not the true averaged failure distribution within the node. Hence, splitting on a nonideal covariate may not be avoidable due to an entanglement between the failure and censoring distributions. We shall see this clearly from a toy example in the next section.

40.6.2 A Motivating Example

To demonstrate the problem within forests, consider the example presented in [6], which we duplicated under slightly

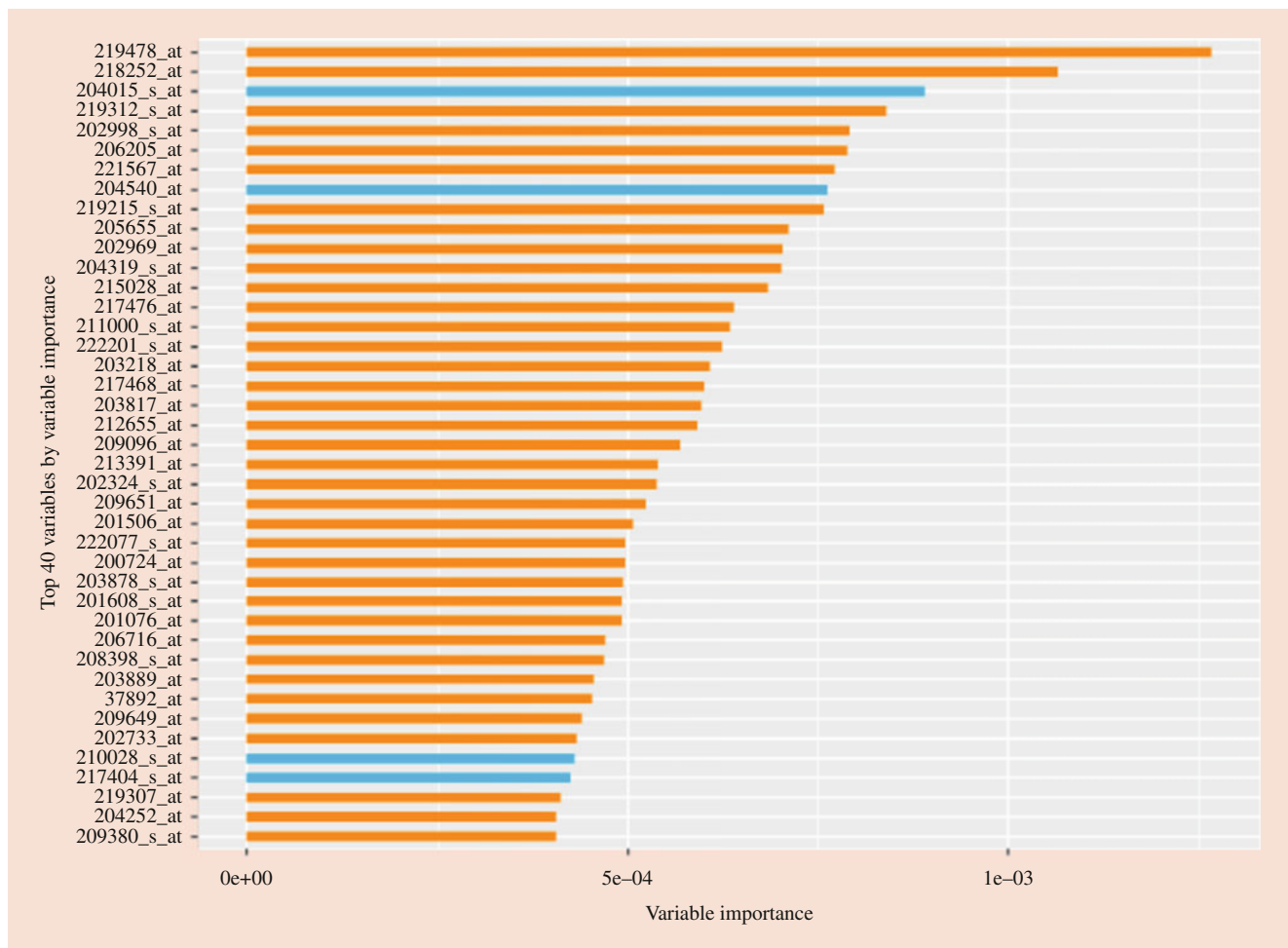


Fig. 40.6 Top 40 variables by variable importance, plotted in order. The *blue bars* are the only ones included in the 76 variable gene signature found by Wang et al. [52]

different settings. Consider $n = 1000$ and $p = 3$. The covariate matrix \mathbf{X} follows a multivariate normal distribution with mean 0 and the covariance between $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ is 0.8, while $\mathbf{X}^{(3)}$ is independent of both $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$. The time until event, T , follows $\text{Exponential}(\exp(1.25\mathbf{X}^{(1)} + \mathbf{X}^{(3)} - 2))$. Hence $\mathbf{X}^{(1)}$ contributes a stronger signal than $\mathbf{X}^{(3)}$. For better understanding, we consider the split at the root node. As such, ideally, the random forest would always split on $\mathbf{X}^{(1)}$ first.

We use two types of censoring distributions. In the first scenario, censoring time is distributed as $\text{Exponential}(\exp(-2))$, so the censoring distribution is independent of all the variables in \mathbf{X} . In the second scenario, the censoring distribution is $\text{Exponential}(\exp(3\mathbf{X}^{(2)} - 2))$, so the censoring distribution is no longer independent of the signal variables. In both cases, approximately 50% of the observations are censored.

The results of our simulation using the log-rank test are displayed in Table 40.1. When the censoring is independent of the signal variables ($\mathbf{X}^{(1)}$ and $\mathbf{X}^{(3)}$), the random forest correctly splits on $\mathbf{X}^{(1)}$ first approximately 86% of the time.

However, when the censoring distribution depends on $\mathbf{X}^{(2)}$ and therefore is no longer independent of $\mathbf{X}^{(1)}$, the random forest splits on $\mathbf{X}^{(1)}$ first in only 34% of cases. It splits first on $\mathbf{X}^{(2)}$ in 56% of the cases, when, ideally, we would not split on $\mathbf{X}^{(2)}$ at all. It is clear that the censoring distribution biases the choice of splitting variables. The consequence of this bias on the consistency of survival trees is much more involved as the entire tree structure can be altered by the censoring distribution. Cui et al. [6] developed a concentration bound on the difference between the accurately predicted hazard function (what we could find with complete data and an unbiased model) and the censoring contaminated predicted hazard function.

40.6.3 Some Solutions for Bias Correction

There have been multiple proposed adjustments to the survival random forest algorithm to deal with the bias. For example, the inverse probability of censoring weights is

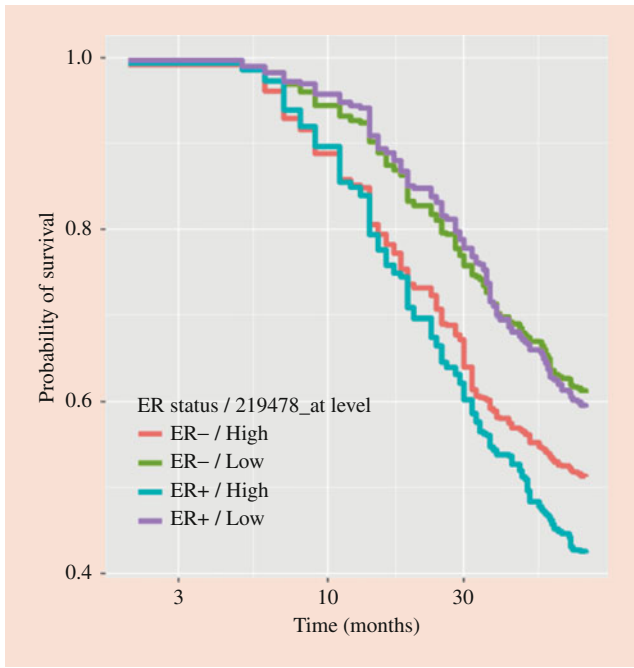


Fig. 40.7 Out-of-bag predicted survival functions for four selected subjects with high or low values of gene “219478_at” and ER+ or ER–

Table 40.1 Probability of selecting each variable in the first split

Censoring mechanism	$X^{(1)}$	$X^{(2)}$	$X^{(3)}$
Independent of signal variables	0.863	0.048	0.089
Not independent of signal variables	0.340	0.562	0.098

extensively considered in the literature [54,59]. For example, [25] proposed utilizing censoring unbiased transformations (CUTs). In particular, they developed a generalized doubly robust transformation that is a CUT when either the survival distribution or the censoring distribution is correctly specified but not necessarily both. They used the CUT to develop a loss function that, when no censoring is present, reduces to a full data function. Then the loss function is used as a splitting rule to create a censoring unbiased regression tree that can then be used for survival random forests.

However, CUTs work in the case where expected values are the focus of analysis. We have seen an example earlier in Sect. 40.2.1 where the expected values were not sufficient to determine the difference between two distributions. To help with bias and deal with the whole distribution (rather than only expected values), we consider a different type of correction. Cui et al. [6] proposed tackling the bias by using a weighted Nelson-Aalen estimator, which is a variation on (40.14):

$$\widehat{\Lambda}(t, \mathbf{x}_0) = \sum_{k:t_k \leq t} \frac{\sum_{i=1}^n \mathbb{1}(\delta_i = 1) \mathbb{1}(y_i = t_k) K(\mathbf{x}_0, \mathbf{x}_i) / [1 - \widehat{G}(t_k, \mathbf{x}_i)]}{\sum_{i=1}^n \mathbb{1}(y_i \geq t_k) K(\mathbf{x}_0, \mathbf{x}_i) / [1 - \widehat{G}(t_k, \mathbf{x}_i)]} \quad (40.27)$$

As in the calculation of the Brier score (40.24), $\widehat{G}(t_k, \mathbf{x}_i)$'s for consistency are estimates of the conditional censoring distribution, which can be estimated by a separate model, or simply plug-in the marginal censoring distributing estimation. Cui et al. [6] estimated and compared the inverse probability weighted hazard functions with the weights $\delta_i / \{1 - \widehat{G}(t_k | \mathbf{x}_i)\}$ at each internal node, which accounts for the bias caused by censoring and allows for the choice of the two most divergent child nodes. This type of splitting rule, like the supremum log-rank test (40.13), compares the survival functions rather than the expected survival time.

Another appeal of this bias-corrected survival forest is in high-dimensional settings. Cui et al. [6] showed that a survival forest model with a marginal splitting rule has the rate of convergence depending on the total number of variables included in both failure time and censoring time. While [6] developed their results under a general framework where the covariates are allowed to be weakly correlated, their approach directly extends to the case where all the variables in \mathbf{x} are independent. In such cases, the biasedness is not due to correlated variables but the entanglement between failure time and censoring time marginally. For example, if the censoring time and failure time depend on a shared variable, all the other variables involved in the censoring distribution may play a role in the limiting distribution of (40.14). Hence, there is no guarantee of splitting only on the failure variables, especially when the dimensionality is high. The new type of survival forests in [6] that leverages (40.27) untangles the failure and censoring distributions and improves the rate of convergence of tree model so that the rate depends only on the number of important variables that define the failure distribution.

40.7 Theoretical Properties of Random Forest

Theoretical analysis of random forests has been an active research topic in recent years. Under assumptions of the additive model, [47] showed the consistency of random forests. One of the earlier attempts was made by Lin and Jeon [60], which formulates random forests as a nearest-neighbor model. An interesting insight provided by this paper is that a random forest defines its neighborhood adaptively. For a variable with a stronger signal, it will be repeatedly used as the splitting rule, making the width of the variable in a terminal node much smaller than other variables [43]. This partition resembles the properties of multidimensional kernel estimators with adaptive bandwidth selection [61]. For random forests, we can easily derive the induced kernel distance as an extension of (40.15)

$$K(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{B} \sum_{b=1}^B \sum_{u \in \mathcal{U}^b} \mathbb{1}(\mathbf{x}_i \in A_u^b) \mathbb{1}(\mathbf{x}_j \in A_u^b), \quad (40.28)$$

which can be used for a kernel version of the prediction [38]. Further exploiting the properties of this induced kernel is of great interest. The technique may be universally applicable to many types of random forest models and shed light on proving the consistency of the method.

Another topic of random forests is to quantify its variation. In statistics, this quantification is an important issue because we may be able to provide a valid inference of the estimated survival function and draw conclusions. For example, in precision medicine [62, 63], one has to compare the survival function estimations that resulted from applying two potential treatments and select the better one [64–66]. Being able to quantify the confidence intervals of the estimations is practically useful. To this end, recent literature proposes to use jackknife [67, 68] or U-statistics [69]. U-statistic [70] is a classical tool that provides asymptotic normality of an averaged estimator based on all possible subsampling combinations. The particular version applicable to the random forest is incomplete [69, 71] in the sense that a relatively small number (`nrtrees`) of such subsamples are used. With suitable conditions, a normality result can be established, and the variance can be estimated by deriving the leading term in the Hoeffding decomposition. However, a relatively restrictive condition is that the subsample size must be smaller than \sqrt{n} . This restriction is a limitation in practice since a small sample size in each tree will lead to significant bias. Being able to relax these conditions while not drastically increasing the computational cost is a challenging statistical issue.

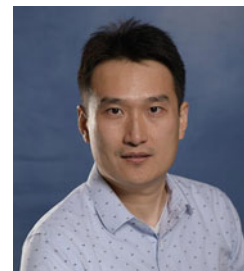
40.8 Supplementary Material

All examples provided in this chapter are created using the R [72]. Source code for the examples is available at <https://github.com/sarahef2/springer-surv>. The RLT package is available at <https://github.com/teazrq/RLT>.

References

1. Fleming, T.R., Harrington, D.P.: Counting Processes and Survival Analysis, vol. 169. Wiley, Hoboken (2011)
2. Nelson, W.: Hazard plotting for incomplete failure data (multiply censored data plotting on various type hazard papers for engineering information on time to failure distribution). *J. Qual. Technol.* **1**, 27–52 (1969)
3. Aalen, O.: Nonparametric inference for a family of counting processes. *Ann. Stat.* **6**(4), 701–726 (1978)
4. Altshuler, B.: Theory for the measurement of competing risks in animal experiments. *Math. Biosci.* **6**, 1–11 (1970)
5. Kaplan, E.L., Meier, P.: Nonparametric estimation from incomplete observations. *J. Am. Stat. Assoc.* **53**(282), 457–481 (1958)
6. Cui, Y., Zhu, R., Zhou, M., Kosorok, M.: Consistency of survival tree and forest models: splitting bias and correction. *Statistica Sinica*, **32**(3), 1245–1267.
7. Cox, D.R.: Regression Models and Life-Tables. *J. R. Stat. Soc. Ser. B Methodol.* **34**(2), 187–220 (1972)
8. Wei, L.-J.: The accelerated failure time model: a useful alternative to the cox regression model in survival analysis. *Stat. Med.* **11**(14–15), 1871–1879 (1992)
9. Kalbfleisch, J.D., Prentice, R.L.: The Statistical Analysis of Failure Time Data. Wiley, New York (1980). ISBN 9780471055198
10. Bennett, S.: Analysis of survival data by the proportional odds model. *Stat. Med.* **2**(2), 273–277 (1983)
11. Vaupel, J.W., Manton, K.G., Stallard, E.: The impact of heterogeneity in individual frailty on the dynamics of mortality. *Demography* **16**(3), 439–454 (1979)
12. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. Taylor & Francis Group, LLC, Boca Raton (1984). ISBN 9781315139470
13. Therneau, T., Atkinson, B.: rpart: Recursive partitioning and regression trees (2019). <https://CRAN.R-project.org/package=rpart>. R package version 4.1-15
14. Hothorn, T., Zeileis, A.: partykit: A modular toolkit for recursive partitioning in R. *J. Mach. Learn. Res.* **16**, 3905–3909 (2015). <http://jmlr.org/papers/v16/hothorn15a.html>
15. Segal, M.R.: Regression trees for censored data. *Biometrics* **44**(1), 35–47 (1988)
16. Loh, W.-Y.: Survival modeling through recursive stratification. *Comput. Stat. Data Anal.* **12**(3), 295–313 (1991)
17. Ahn, H., & Loh, W.-Y.: Tree-Structured Proportional Hazards Regression Modeling. *Biometrics*, **50**(2), 471–485 (1994).
18. Su, X., Fan, J.: Multivariate survival trees: a maximum likelihood approach based on frailty models. *Biometrics* **60**(1), 93–99 (2004)
19. Molinaro, A.M., Dudoit, S., Van der Laan, M.J.: Tree-based multivariate regression and density estimation with right-censored data. *J. Multivar. Anal.* **90**(1), 154–177 (2004)
20. Bou-Hamad, I., Larocque, D., Ben-Ameur, H.: A review of survival trees. *Statistics Surveys* **5**, 44–71 (2011)
21. Fleming, T.R., O’Sullivan, M., Harrington, D.P.: Supremum versions of the log-rank and generalized Wilcoxon statistics. *J. Am. Stat. Assoc.* **82**(397), 312–320 (1987). ISSN 1537274X. doi: 10.1080/01621459.1987.10478435
22. Kosorok, M.R., Lin, C.-y.: The Versatility of Function-Indexed Weighted Log-Rank Statistics. *J. Am. Stat. Assoc.* **94**(445), 320–332 (1999)
23. Ishwaran, H., Kogalur, U.B., Blackstone, E.H., Lauer, M.S.: Random survival forests. *Ann. Appl. Stat.* **2**(3), 841–860 (2008)
24. Zhu, R., Kosorok, M.R.: Recursively imputed survival trees. *J. Am. Stat. Assoc.* **107**(497), 331–340 (2012)
25. Steingrimsson, J.A., Diao, L., Strawderman, R.L.: Censoring unbiased regression trees and ensembles. *J. Am. Stat. Assoc.* **114**(525), 370–383 (2019)
26. Hastie, T., Tibshirani, R., Friedman, J., Franklin, J.: The elements of statistical learning: data mining, inference and prediction. *Math. Intell.* **27**(2), 83–85 (2005)
27. Nadaraya, E.A.: On estimating regression. *Theory Probab. Appl.* **9**(1), 141–142 (1964)
28. Dabrowska, D.M.: Uniform Consistency of the Kernel Conditional Kaplan-Meier Estimate. *Ann. Stat.* **17**(3), 1157–1167, (1989)
29. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
30. Breiman, L.: Bagging predictors. *Mach. Learn.* **24**(2), 123–140 (1996)
31. Amit, Y., Geman, D.: Shape quantization and recognition with randomized trees. *Neural Comput.* **9**(7), 1545–1588 (1997)
32. Ho, T.K.: The random subspace method for constructing decision forests. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(8), 832–844 (1998)
33. Hothorn, T., Lausen, B., Benner, A., Radespiel-Tröger, M.: Bagging survival trees. *Stat. Med.* **23**(1), 77–91 (2004)

34. Meinshausen, N.: Quantile regression forests. *J. Mach. Learn. Res.* **7**, 983–999 (2006)
35. Shi, T., Horvath, S.: Unsupervised learning with random forest predictors predictors. *J. Comput. Graph. Stat.* **15**(1), 118–138 (2006). doi: 10.1198/106186006X94072.
36. Liaw, A., Wiener, M., et al.: Classification and regression by randomforest. *R News* **2**(3), 18–22 (2002)
37. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. *Mach. Learn.* **63**(1), 3–42 (2006)
38. Scornet, E.: Random forests and kernel methods. *IEEE Trans. Inf. Theory* **62**(3), 1485–1500 (2016)
39. Athey, S., Tibshirani, J., Wager, S., et al.: Generalized random forests. *Ann. Stat.* **47**(2), 1148–1178 (2019)
40. Liaw, A., Wiener, M.: Classification and regression by randomforest. *R News* **2**(3), 18–22 (2002). <https://CRAN.R-project.org/doc/Rnews/>
41. Ishwaran, H., Kogalur, U.B.: Fast unified random forests for survival, regression, and classification (rf-src) (2019). <https://cran.r-project.org/package=randomForestSRC>. R package version 2.9.1
42. Wright, M.N., Ziegler, A.: ranger: A fast implementation of random forests for high dimensional data in C++ and R. *J. Stat. Softw.* **77**(1), 1–17 (2017). doi: 10.18637/jss.v077.i01
43. Zhu, R., Zeng, D., Kosorok, M.R.: Reinforcement learning trees. *J. Am. Stat. Assoc.* **110**(512), 1770–1784 (2015)
44. Harrell, F.E., Califf, R.M., Pryor, D.B., Lee, K.L., Rosati, R.A.: Evaluating the Yield of Medical Tests. *JAMA* **247**(18), 2543–2546 (1982)
45. Graf, E., Schmoor, C., Sauerbrei, W., Schumacher, M.: Assessment and comparison of prognostic classification schemes for survival data. *Stat. Med.* **18**(17–18), 2529–2545 (1999). ISSN 0277-6715
46. Brier, G.W.: Verification of forecasts expressed in terms of probability. *Mon. Weather Rev.* **78**(1), 1–3 (1950)
47. Scornet, E., Biau, G., Vert, J.-P.: Consistency of random forests. *Ann. Stat.* **43**(4), 1716–1741 (2015)
48. Strobl, C., Boulesteix, A.L., Zeileis, A., Hothorn, T.: Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinf.* **8**(1), 1–21 (2007). ISSN 14712105. doi: 10.1186/1471-2105-8-25
49. Strobl, C., Boulesteix, A.L., Kneib, T., Augustin, T., Zeileis, A.: Conditional variable importance for random forests. *BMC Bioinf.* **9**, 1–11 (2008). ISSN 14712105. doi: 10.1186/1471-2105-9-307
50. Ishwaran, H., Kogalur, U.B., Chen, X., Minn, A.J.: Random survival forests for high-dimensional data. *Stat. Anal. Data Min.* **4**(1), 115–132 (2011). ISSN 19321864. doi: 10.1002/sam.10103. <http://doi.wiley.com/10.1002/sam.10103>
51. Milborrow, S.: rpart.plot: Plot ‘rpart’ models: An enhanced version of ‘plot.rpart’ (2019). <https://CRAN.R-project.org/package=rpart.plot>. R package version 3.0.8
52. Wang, Y., Klijn, J.G.M., Zhang, Y., Sieuwerts, A.M., Look, M.P., Yang, F., Talantov, D., Timmermans, M., Meijer-Van Gelder, M.E., Yu, J., Jatke, T., Berns, E.M.J.J., Atkins, D., Foekens, J.A.: Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *Lancet* **365**, 671–679 (2005). ISSN 01406736. doi: 10.1016/S0140-6736(05)70933-8
53. Edgar, R., Domrachev, M., Lash, A.E.: Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Res.* **30**(1), 207–210 (2002)
54. Hothorn, T., Bühlmann, P., Dudoit, S., Molinaro, A., Van Der Laan, M.J.: Survival ensembles. *Biostatistics* **7**(3), 355–373 (2005)
55. Buckley, J., James, I.: Linear regression with censored data. *Biometrika* **66**(3), 429–436 (1979)
56. Breslow, N., Crowley, J.: A large sample study of the life table and product limit estimates under random censorship. *Ann. Stat.* **2**(3), 437–453 (1974)
57. Gill, R.D.: Censoring and stochastic integrals. *Statistica Neerlandica* **34**(2), 124–124 (1980)
58. Zhou, M., et al.: Some properties of the kaplan-meier estimator for independent nonidentically distributed random variables. *Ann. Stat.* **19**(4), 2266–2274 (1991)
59. Van der Laan, M.J., Laan, M.J., Robins, J.M.: Unified Methods for Censored Longitudinal Data and Causality. Springer, Berlin (2003)
60. Lin, Y., Jeon, Y.: Random forests and adaptive nearest neighbors. *J. Am. Stat. Assoc.* **101**(474), 578–590 (2006)
61. Li, Q., Lin, J., Racine, J.S.: Optimal bandwidth selection for non-parametric conditional distribution and quantile functions. *J. Bus. Econ. Stat.* **31**(1), 57–65 (2013)
62. Foster, J.C., Taylor, J.M.G., Ruberg, S.J.: Subgroup identification from randomized clinical trial data. *Stat. Med.* **30**(24), 2867–2880 (2011)
63. Wager, S., Athey, S.: Estimation and inference of heterogeneous treatment effects using random forests. *J. Am. Stat. Assoc.* **113**(523), 1228–1242 (2018)
64. Zhao, Y.-Q., Zeng, D., Laber, E.B., Song, R., Yuan, M., Kosorok, M.R.: Doubly robust learning for estimating individualized treatment with censored data. *Biometrika* **102**(1), 151–168 (2014)
65. Zhu, R., Zhao, Y.Q., Chen, G., Ma, S., Zhao, H.: Greedy outcome weighted tree learning of optimal personalized treatment rules. *Biometrics* **73**(2), 391–400 (2017)
66. Cui, Y., Zhu, R., Kosorok, M.: Tree based weighted learning for estimating individualized treatment rules with censored data. *Electron. J. Statist.* **11**(2), 3927–3953 (2017). doi: 10.1214/17-EJS1305. <https://doi.org/10.1214/17-EJS1305>
67. Sexton, J., Laake, P.: Standard errors for bagged and random forest estimators. *Comput. Stat. Data Anal.* **53**(3), 801–811 (2009)
68. Wager, S., Hastie, T., Efron, B.: Confidence intervals for random forests: the jackknife and the infinitesimal jackknife. *J. Mach. Learn. Res.* **15**(1), 1625–1651 (2014)
69. Mentch, L., Hooker, G.: Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. *J. Mach. Learn. Res.* **17**(1), 841–881 (2016)
70. Hoeffding, W.: A class of statistics with asymptotically normal distribution. In: *Breakthroughs in Statistics*, pp. 308–334. Springer, Berlin (1992)
71. Lee, A.J.: *U-statistics: Theory and Practice*. Routledge, London (2019)
72. R Core Team. *R: A Language and Environment for Statistical Computing* (2019). <https://www.R-project.org>



Ruqing Zhu is an Associate Professor at the Department of Statistics at UIUC. He obtained his Ph.D from UNC Chapel Hill in 2013 and completed his postdoc training at Yale University in 2015. His expertise is in statistical methodology and theory of machine learning methods and their applications to biomedical data. Currently, his research focuses on personalized medicine, random forests, reinforcement learning and survival analysis.



Sarah E. Formentini received her Ph.D. in Statistics at the University of Illinois at Urbana-Champaign in 2022. Her research focused on survival analysis and random forests, including survival data integration and survival random forest variation estimation.



Yifan Cui received his Ph.D. from University of North Carolina at Chapel Hill in 2018. He is currently a tenure track faculty member at the Center for Data Science in Zhejiang University (ZJU). He has worked at Wharton Statistics Department, University of Pennsylvania as a postdoctoral researcher and Department of Statistics and Data Science, National University of Singapore as an assistant professor. His main research interests include nonparametric and semiparametric statistics, random forests, causal inference, precision medicine, survival analysis, generalized fiducial inference, and foundations of statistics.



Probability Inequalities for High-Dimensional Time Series Under a Triangular Array Framework

41

Fang Han and Wei Biao Wu

Contents

41.1	The Measure of Dependence	849
41.2	Probability and Moment Inequalities Under Dependence	851
41.2.1	Sample Sum for Scalars with $d_n = 1$	852
41.2.2	Sample Sum for Random Vectors with $d_n \geq 1$	854
41.2.3	Sample Sum for Random Matrices with $d_n \geq 1$	856
41.2.4	U- and V-Statistics.....	857
41.3	A Cautionary Example	858
41.4	Statistical Applications	861
References	862

Keywords

Mixing conditions · Weak dependence measures · τ -mixing · Functional dependence measure · High dimension · Time series analysis

41.1 The Measure of Dependence

We first introduce the mixing conditions defined on σ -fields. Fix the probability space as $(\Omega, \mathcal{F}, \mathbb{P})$. For any two σ -fields \mathcal{A}, \mathcal{B} belonging to \mathcal{F} , define the following four measures of dependence between \mathcal{A} and \mathcal{B} (cf. Chapter 3, [10]):

$$\alpha(\mathcal{A}, \mathcal{B}) := \sup_{A \in \mathcal{A}, B \in \mathcal{B}} \left| \mathbb{P}(A \cap B) - \mathbb{P}(A)\mathbb{P}(B) \right|,$$

$$\beta(\mathcal{A}, \mathcal{B}) := \sup \frac{1}{2} \sum_{i=1}^I \sum_{j=1}^J \left| \mathbb{P}(A_i \cap B_j) - \mathbb{P}(A_i)\mathbb{P}(B_j) \right|,$$

$$\phi(\mathcal{A}, \mathcal{B}) := \sup_{A \in \mathcal{A}, B \in \mathcal{B}, \mathbb{P}(A) > 0} \left| \mathbb{P}(B | A) - \mathbb{P}(B) \right|,$$

$$\rho(\mathcal{A}, \mathcal{B}) := \sup \left\{ \left| \text{Corr}(f, g) \right|, f \in \mathcal{L}_{\mathbb{R}}^2(\mathcal{A}), g \in \mathcal{L}_{\mathbb{R}}^2(\mathcal{B}) \right\},$$

Abstract

Study of time series data often involves measuring the strength of temporal dependence, on which statistical properties like consistency and central limit theorem are built. Historically, various dependence measures have been proposed. In this note, we first survey some of the most well-used dependence measures as well as various probability and moment inequalities built upon them under a high-dimensional triangular array time series setting. We then argue that this triangular array setting will pose substantially new challenges to the verification of some dependence conditions. In particular, “textbook results” could now be misleading and hence are recommended to be used with caution.

where the supremum in the definition of $\beta(\mathcal{A}, \mathcal{B})$ is taken over all pairs of partitions $\{A_1, \dots, A_I\}$ and $\{B_1, \dots, B_J\}$ such that $A_i \in \mathcal{A}$ and $B_j \in \mathcal{B}$ for all i, j , and for any $p \in [1, \infty]$, let $\mathcal{L}_{\mathbb{R}}^p(\mathcal{A})$ represent the family of all real valued, \mathcal{A} -measurable random variables X on Ω such that $\|X\|_{L_p} := (\mathbb{E}|X|^p)^{1/p} < \infty$. We refer to [9] for basic properties and historical developments on these dependence measures.

Now let’s consider a (not necessarily stationary) time series $\{X_t \in \mathbb{R}^d\}_{t \in \mathbb{Z}}$ with \mathbb{Z} and \mathbb{R}^d representing the sets of all integers and all d -dimensional real vectors. For each “time gap” $m = 1, 2, \dots$, with the above dependence measures, we are now ready to define the following four mixing coefficients that appear frequently in literature:

F. Han (✉)
University of Washington, Seattle, WA, USA
e-mail: fanghan@uw.edu

W. B. Wu
University of Chicago, Chicago, IL, USA
e-mail: wbwu@galton.uchicago.edu

$$\begin{aligned} \alpha(\{X_t\}_{t \in \mathbb{Z}}; m) &:= \sup_{j \in \mathbb{Z}} \alpha(\sigma(\{X_t\}_{t \leq j}), \sigma(\{X_t\}_{t \geq j+m})), \\ \beta(\{X_t\}_{t \in \mathbb{Z}}; m) &:= \sup_{j \in \mathbb{Z}} \beta(\sigma(\{X_t\}_{t \leq j}), \sigma(\{X_t\}_{t \geq j+m})), \\ \phi(\{X_t\}_{t \in \mathbb{Z}}; m) &:= \sup_{j \in \mathbb{Z}} \phi(\sigma(\{X_t\}_{t \leq j}), \sigma(\{X_t\}_{t \geq j+m})), \\ \rho(\{X_t\}_{t \in \mathbb{Z}}; m) &:= \sup_{j \in \mathbb{Z}} \rho(\sigma(\{X_t\}_{t \leq j}), \sigma(\{X_t\}_{t \geq j+m})). \end{aligned}$$

Here for any random variable X , $\sigma(X)$ is understood to be the σ -field generated by X . A review of the history of these mixing coefficients can be found in Section 2.1 in [9]. We also refer readers to the books of [10, 16], and [39].

The above mixing coefficients are defined on σ -fields and are usually difficult to be explicitly calculated in practice, though when the model is fixed, asymptotic bounds on coefficients can be derived for many time series models and have been established in many works. This is one of the reasons to define weak dependence measures that are often much easier to calculate. In the following we introduce several of the most well-used ones.

Bickel and Bühlmann [8] and Doukhan and Louhichi [17] introduced a notion of weak dependence that facilitates explicit calculation of the independence strength between “past” and “future” without resorting to the latent σ -fields. They could be roughly understood as upper bounding

$$\text{Cov}(f(\text{“past”}), g(\text{“future”}))$$

by the gap between “past” and “future” as well as some parameters of the functions f and g . In detail, for a function $g : (\mathbb{R}^d)^u \rightarrow \mathbb{R}$, let’s define

$$\text{Lip}_\delta g := \sup \left\{ \frac{|g(x_1, \dots, x_u) - g(y_1, \dots, y_u)|}{\delta((x_1, \dots, x_u), (y_1, \dots, y_u))} : (x_1, \dots, x_u) \neq (y_1, \dots, y_u) \right\},$$

where $\delta(\cdot)$ represents a certain metric on the real vector space. Denote $\Lambda_\delta := \{g : (\mathbb{R}^d)^u \rightarrow \mathbb{R} \text{ for some } u : \text{Lip}_\delta g < \infty\}$ and $\Lambda_\delta^{(1)} := \{g \in \Lambda_\delta : \|g\|_\infty \leq 1\}$ with $\|g\|_\infty := \sup_x |g(x)|$. In the following, \mathbb{N} represents the set of all natural numbers.

Definition 1.1 ([17, 18]) The process $\{X_t\}_{t \in \mathbb{Z}}$ is $(\Lambda_\delta^{(1)}, \psi, \zeta)$ -weakly dependent if and only if there exists a function $\psi : \mathbb{R}_+^2 \times \mathbb{N}^2 \rightarrow \mathbb{R}_+$ and a sequence $\zeta = \{\zeta(n)\}_{n \geq 0}$ decreasing to 0 as n goes to infinity, such that for any $g_1, g_2 \in \Lambda_\delta^{(1)}$ with $g_1 : (\mathbb{R}^d)^u \rightarrow \mathbb{R}, g_2 : (\mathbb{R}^d)^v \rightarrow \mathbb{R}, u, v \in \mathbb{N}$, and any u -tuple (s_1, \dots, s_u) and any v -tuple (t_1, \dots, t_v) with $s_1 \leq \dots \leq s_u < t_1 \leq \dots \leq t_v$, the following inequality is satisfied:

$$\begin{aligned} & \left| \text{Cov} \left\{ g_1(x_{s_1}, \dots, x_{s_u}), g_2(x_{t_1}, \dots, x_{t_v}) \right\} \right| \\ & \leq \psi(\text{Lip}_\delta g_1, \text{Lip}_\delta g_2, u, v) \zeta(t_1 - s_u). \end{aligned}$$

Important examples of $(\Lambda_\delta^{(1)}, \psi, \zeta)$ -weakly dependent processes include θ -, η -, κ -, and λ -dependences, which are listed in Table 41.1. They correspond to different choices of the function ψ . Similar to the mixing coefficients, the sequence ζ describes the degree of dependence over the process.

Later, in [13] and [14], the authors introduced a new set of dependence measures that, instead of putting focus on the covariance structure, highlights the intrinsic “coupling” property of the sequence. This note shall focus on one important member in this family, the τ -dependence. Consider a general probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and a random variable X taking value in a Polish space $(\mathcal{X}, \|\cdot\|_{\mathcal{X}})$ endowed with a norm $\|\cdot\|_{\mathcal{X}}$ and satisfying $\|X - x_0\|_{\mathcal{X}} \|L_1 < \infty$ for some $x_0 \in \mathcal{X}$. Consider a σ -field $\mathcal{A} \subset \mathcal{F}$. The τ -measure of dependence between X and \mathcal{A} is defined to be

$$\begin{aligned} \tau(\mathcal{A}, X; \|\cdot\|_{\mathcal{X}}) &= \left\| \sup_{g \in \Lambda(\|\cdot\|_{\mathcal{X}})} \left\{ \int g(x) \mathbb{P}_{X|\mathcal{A}}(dx) \right. \right. \\ & \quad \left. \left. - \int g(x) \mathbb{P}_X(dx) \right\} \right\|_{L_1}, \end{aligned}$$

where \mathbb{P}_X and $\mathbb{P}_{X|\mathcal{A}}$ represent the distribution and the conditional distributions of X and X given \mathcal{A} and $\Lambda(\|\cdot\|_{\mathcal{X}})$ stands for the set of 1-Lipschitz functions from \mathcal{X} to \mathbb{R} with respect to the norm $\|\cdot\|_{\mathcal{X}}$.

The following theorem, extracted from [13] and [15], characterizes the intrinsic “coupling property” of the τ -measure of dependence and, as a matter of fact, gives an alternative definition of τ -measure that is usually easier to use.

Theorem 1.1 (Lemma 3 in [13], Lemma 5.3 in [15]) Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, \mathcal{A} be a σ -field of \mathcal{F} , and X be a random variable with values in a Polish space $(\mathcal{X}, \|\cdot\|_{\mathcal{X}})$. If Y is a random variable distributed as X and independent of \mathcal{A} , then

$$\tau(\mathcal{A}, X; \|\cdot\|_{\mathcal{X}}) \leq \mathbb{E} \|X - Y\|_{\mathcal{X}}.$$

Table 41.1 Important examples of weak dependence

θ -dependence:	$\psi(\text{Lip}_\delta g_1, \text{Lip}_\delta g_2, u, v) = v \text{Lip}_\delta g_2$
η -dependence:	$\psi(\text{Lip}_\delta g_1, \text{Lip}_\delta g_2, u, v) = u \text{Lip}_\delta g_1 + v \text{Lip}_\delta g_2$
κ -dependence:	$\psi(\text{Lip}_\delta g_1, \text{Lip}_\delta g_2, u, v) = uv \text{Lip}_\delta g_1 \text{Lip}_\delta g_2$
λ -dependence:	$\psi(\text{Lip}_\delta g_1, \text{Lip}_\delta g_2, u, v) = u \text{Lip}_\delta g_1 + v \text{Lip}_\delta g_2 + uv \text{Lip}_\delta g_1 \text{Lip}_\delta g_2$

Assume that $\int \|x - x_0\|_{\mathcal{X}} \mathbb{P}_X(\mathrm{d}x)$ is finite for any $x_0 \in \mathcal{X}$. Assume that there exists a random variable U uniformly distributed over $[0, 1]$, independent of the sigma-field generated by X and \mathcal{A} . Then there exists a random variable \tilde{X} , measurable with respect to $\mathcal{A} \vee \sigma(X) \vee \sigma(U)$, independent of \mathcal{A} and distributed as X , such that

$$\tau(\mathcal{A}, X; \|\cdot\|_{\mathcal{X}}) = \mathbb{E}\|X - \tilde{X}\|_{\mathcal{X}}.$$

We now apply the notion of τ -dependence to a time series model. Let $\{X_j\}_{j \in J}$ be a set of \mathcal{X} -valued random variables with index set J of finite cardinality. Then define

$$\tau(\mathcal{A}, \{X_j \in \mathcal{X}\}_{j \in J}; \|\cdot\|_{\mathcal{X}}) = \left\| \sup_{g \in \Lambda(\|\cdot\|'_{\mathcal{X}})} \left\{ \int g(x) \mathbb{P}_{\{X_j\}_{j \in J} | \mathcal{A}}(\mathrm{d}x) - \int g(x) \mathbb{P}_{\{X_j\}_{j \in J}}(\mathrm{d}x) \right\} \right\|_{L_1},$$

where $\mathbb{P}_{\{X_j\}_{j \in J}}$ and $\mathbb{P}_{\{X_j\}_{j \in J} | \mathcal{A}}$ represent the distribution of $\{X_j\}_{j \in J}$ and the conditional distribution of $\{X_j\}_{j \in J}$ given \mathcal{A} respectively, and $\Lambda(\|\cdot\|'_{\mathcal{X}})$ stands for the set of 1-Lipschitz functions:

$$\Lambda(\|\cdot\|'_{\mathcal{X}}) := \left\{ f : \underbrace{\mathcal{X} \times \dots \times \mathcal{X}}_{\text{Card}(J)} \rightarrow \mathbb{R}; \right. \\ \left. f \text{ is 1-Lipschitz with respect to } \|\cdot\|'_{\mathcal{X}} \right\}$$

with $\|x\|'_{\mathcal{X}} := \sum_{j \in J} \|x_j\|_{\mathcal{X}}$ for any $x = (x_1, \dots, x_J) \in \mathcal{X}^{\text{Card}(J)}$.

Using these concepts, for a time series $\{X_t\}_{t \in \mathbb{Z}}$, it is ready to define measure of temporal correlation strength as

$$\tau(\{X_t\}_{t \in \mathbb{Z}}; m, \|\cdot\|_{\mathcal{X}}) := \sup_{i > 0} \max_{1 \leq \ell \leq i} \ell^{-1} \sup \left\{ \tau(\sigma(X_{-\infty}^a), \{X_{j_1}, \dots, X_{j_\ell}\}; \|\cdot\|_{\mathcal{X}}), a \right. \\ \left. + m \leq j_1 < \dots < j_\ell \right\},$$

where the inner supremum is taken over all $a \in \mathbb{Z}$ and all ℓ -tuples (j_1, \dots, j_ℓ) .

In the end, let's consider $\{X_t\}_{t \in \mathbb{Z}}$ to be a real stationary causal process of the form

$$X_i = g(\dots, \epsilon_{i-1}, \epsilon_i), \tag{41.1}$$

with $\{\epsilon_i\}_{i \in \mathbb{Z}}$ an independent and identically distributed (i.i.d.) sequence and $g(\cdot)$ a measurable function such that the above time series model is properly defined. In [42], the author introduced the functional dependence measure, as manifested below.

Definition 1.2 (Functional Dependence Measure, [42]) Let $\{\epsilon_i, \epsilon'_j\}_{i,j \in \mathbb{Z}}$ be i.i.d. random variables. Let $X'_m := g(\dots, \epsilon_{-2}, \epsilon_{-1}, \epsilon'_0, \epsilon_1, \dots, \epsilon_m)$. The functional dependence measure with regard to the L_p norm is defined to be

$$\theta_{m,p} := \|X_m - X'_m\|_{L_p}$$

with the tail sum $\Theta_{m,p} := \sum_{k=m}^{\infty} \theta_{k,p}$.

The functional dependence measure $\theta_{m,p}$ is flexible and easy to compute in many applications; we refer the readers of interest to [43] for a systematic review. In addition, given the data generating mechanism g , one could numerically compute functional dependence measures by Monte Carlo simulations. In contrast, numeric computation of other dependence measures can be highly nontrivial due to their definitions.

We also mention a connection between physical dependence and τ -dependence. As is apparent by comparing Theorem 1.1 with Definition 1.2, τ -dependence and functional dependence measure are interestingly intrinsically connected. In particular, they are both adaptable to a notion of coupling. However, as noted in Remark 3.1 in [15], coupling in functional dependence is given in [13] with all elements in the past, while in [42] with only element in the past.

41.2 Probability and Moment Inequalities Under Dependence

Probability and moment inequalities play an important role in studying the statistical properties of estimators of parameters in statistical models. They are key in high-dimensional statistical theory, which is by its nature nonasymptotic. Of particular importance are those that give rise to efficient control of tail deviations, namely, higher-order moment and exponential-type inequalities. In this section we will give a brief review of some developed inequalities for time series, which are promising to be applied to the analysis of high-dimensional time series data. For this, this note is restricted to those built on the weak dependence measures introduced in Sect. 41.1, while those built on other structures like Markov chains or martingales, though related, shall not be covered.

Before diving into the details, let's first fix what we mean a high-dimensional time series model. To characterize the impact of dimensionality on the performance of an estimator, it has become well-accepted in literature to model high-dimensional data under a triangular-array setting; see, for example, Section 1 in [21] for a comprehensive illustration. Applied to time series models, the following model will be used throughout the rest of this paper: For each $n \in \mathbb{N}$, let $\{X_{t,n}\}_{t \in \mathbb{Z}}$ denote a d_n -dimensional real time series with $d_n \in \mathbb{N}$ as well as the time series itself depending on n .

For each $n \in \mathbb{N}$, a length of n fragment $\{X_{i,n}\}_{i \in [n]}$, with $[n] := \{1, 2, \dots, n\}$, is observed from the time series $\{X_{t,n}\}_{t \in \mathbb{Z}}$. For different n , a different time series with possibly different dimension is observed. In particular, as n goes to infinity, the dimension of the n fragment time series, d_n , is allowed to increase to infinity as well.

To name one particular example, let's consider the observations $\{X_{i,n}\}_{i \in [n]}$ to be generated from a VAR(1) model, \mathcal{M}_n , that is changing with n :

$$\mathcal{M}_n : \left\{ \{X_{t,n}\}_{t \in \mathbb{Z}} : X_{t,n} = A_n X_{t-1,n} + E_{t,n}, \text{ for all } t \in \mathbb{Z} \right\}.$$

Here A_n is a $d_n \times d_n$ -dimensional transition matrix, and $E_{t,n}$ is a d_n -dimensional vector of error term. The value A_n and the dimension d_n are both allowed to change with n ; e.g., it could be true that

As $n = 1$, a 1-dimensional, length of 1 fragment, $\{X_{1,1}\}$, is observed from the model \mathcal{M}_1 with $A_1 = 0.5$;

As $n = 2$, a 2-dimensional, length of 2 fragment time series, $\{X_{1,2}, X_{2,2}\}$, is observed from the model \mathcal{M}_2 with

$$A_2 = \begin{pmatrix} 0.5 & 0.1 \\ 0.2 & 0.25 \end{pmatrix};$$

As $n = 3$, a 4-dimensional, length of 3 fragment time series, $\{X_{1,3}, X_{2,3}, X_{3,3}\}$, is observed from the model \mathcal{M}_3 with

$$A_3 = \begin{pmatrix} 0.2 & 0 & 0.1 & 0.4 \\ 0.2 & 0.1 & 0.1 & 0.2 \\ 0.1 & 0.2 & 0.3 & 0.1 \\ 0 & 0 & 0 & 0.1 \end{pmatrix};$$

.....

41.2.1 Sample Sum for Scalars with $d_n = 1$

Several of the most essential moment inequalities are centered around the sample sum. In detail, for any $n \in \mathbb{N}$, consider a time series $\{X_{t,n}\}_{t \in \mathbb{Z}}$ and its size- n fragment $\{X_{i,n}\}_{i \in [n]}$. Our aim is to characterize the moment and tail properties for $\sum_{i=1}^n (X_{i,n} - \mathbb{E}X_{i,n})$. Without loss of generality, in the following it is assumed that the time series has margin mean-zero. In this section we are focused on the sample sum $S_n := \sum_{i=1}^n X_{i,n}$ of fixed dimension $d_n = 1$; in the later sections we shall allow d_n to increase to infinity.

To start with, let's first consider the case of linear processes by assuming that $\{X_{t,n}\}_{t \in \mathbb{Z}}$ follows a linear process

$$X_{t,n} = \sum_{j=0}^{\infty} f_{j,n} \epsilon_{t-j,n}, \tag{41.2}$$

with $\{\epsilon_{j,n}\}_{j \in \mathbb{Z}}$ understood to be an i.i.d. scalar sequence with mean zero and $\|\epsilon_{0,n}\|_{L_p} < \infty$ for some $p > 2$, and $f_n := \{f_{j,n}\}$

as a real coefficient sequence satisfying $\|f_n\|_2^2 := \sum_{j=0}^{\infty} f_{j,n}^2 < \infty$. The form (41.2) is very general and includes many famous time series models such as the ARMA processes.

The first result concerns such time series of the particular form (41.2) and is from [44]. It gives a Nagaev-type inequality for linear processes, including both short- and long-range dependence cases.

Theorem 2.1 (Theorem 1, [44]) *Assume the linear process in (41.2). Then the following two statements are true.*

(i) (Short-range dependence) *Let $c_p := 2e^{-p}(p+2)^{-2}$. If $\|f_n\|_1 := \sum_{j=0}^{\infty} |f_{j,n}| < \infty$, then for any $x > 0$ we have*

$$\mathbb{P}(|S_n| \geq x) \leq \left(1 + \frac{2}{p}\right)^p \cdot \frac{n \|f_n\|_1^p \|\epsilon_{0,n}\|_{L_p}^p}{x^p} + 2 \exp\left(-\frac{c_p x^2}{n \|f_n\|_1^2 \|\epsilon_{0,n}\|_{L_2}^2}\right).$$

(ii) (Long-range dependence) *Assume $K_n := \sup_{j \geq 0} |f_{j,n}|(1+j)^\beta < \infty$ for some $1/2 < \beta < 1$. Then there exist constants C_1, C_2 only depending on p and β such that, for all $x > 0$,*

$$\mathbb{P}(|S_n| \geq x) \leq C_1 \frac{n^{1+p(1-\beta)} K_n^p \|\epsilon_{0,n}\|_{L_p}^p}{x^p} + 2 \exp\left(-\frac{C_2 x^2}{n^{3-2\beta} \|\epsilon_{0,n}\|_{L_2}^2 K_n^2}\right).$$

We then move on to the general possibly nonlinear case. The first of such results considers the ϕ -mixing case and is from [14].

Theorem 2.2 (Proposition 5, [14]) *Let $\{X_{t,n}\}_{t \in \mathbb{Z}}$ be a mean-zero stationary sequence of dimension d_n fixed to be 1. Let $\phi_n(m) := \phi(\{X_{t,n}\}_{t \in \mathbb{Z}}; m)$ and $|X_{0,n}| \leq C_n$ for some constant C_n that possibly depends on n . Then, for every $p = 2, 3, \dots$ and any $n \geq 1$, the following inequality holds:*

$$\mathbb{E}|S_n|^p \leq \left(8C_n^2 p \sum_{i=0}^{n-1} (n-i)\phi_n(i)\right)^{p/2}.$$

The next result considers the α - and τ -mixing cases and is from [35].

Theorem 2.3 (Theorem 2, [35]) *Let $\{X_{t,n}\}_{t \in \mathbb{Z}}$ be a stationary mean-zero sequence of dimension d_n fixed to be 1. Suppose that the sequence satisfies either a geometric α -mixing condition:*

$$\alpha(\{X_{t,n}\}_{t \in \mathbb{Z}}; m) \leq \exp(-\gamma_n m), \text{ for } m = 1, 2, \dots$$

or a geometric τ -mixing condition:

$$\tau(\{X_{t,n}\}_{t \in \mathbb{Z}}; m, |\cdot|) \leq \exp(-\gamma_n m), \text{ for } m = 1, 2, \dots$$

with some positive constant γ_n that could depend on n , and there exists a positive constant B_n such that $\sup_{i \geq 1} \|X_{i,n}\|_{L_\infty} \leq B_n$. Then there are positive constants $C_{1,n}$ and $C_{2,n}$ depending only on γ_n such that for all $n \geq 2$ and positive $t < 1/[C_1 B_n (\log n)^2]$, the following inequality holds:

$$\log[\mathbb{E} \exp(tS_n)] \leq \frac{C_{2,n} t^2 (n\sigma_n^2 + B_n^2)}{1 - C_{1,n} t B_n (\log n)^2},$$

where σ_n^2 is defined by

$$\sigma_n^2 := \text{Var}(X_{1,n}) + 2 \sum_{i>1} \left| \text{Cov}(X_{1,n}, X_{i,n}) \right|.$$

We note here that the dependence of $C_{1,n}$ and $C_{2,n}$ on γ_n could be explicitly calculated, as have been made in [6] and [23]; also refer to the later Theorems 2.10 and 2.11.

The next result considers the weak dependence case and is the foundation of dependence measures proposed in [17]. We refer to [17] and [18] for the relation between those weak dependences defined in Definition 1.1 and the following Eqs. (41.3) and (41.4).

Theorem 2.4 (A Slight Modification to Theorem 1 in [18]) Suppose $\{X_{i,n}\}_{i \in [n]}$ are real-valued random variables with mean 0, defined on a common probability space $(\Omega, \mathcal{A}, \mathbb{P})$. Let $\Psi : \mathbb{N}^2 \rightarrow \mathbb{N}$ be one of the four functions defined in Table 41.1. Assume that there exist constants $K_n, M_n, L_{1,n}, L_{2,n} > 0$, $a_n, b_n \geq 0$, and a nonincreasing sequence of real coefficients $\{\rho_n(i)\}_{i \geq 0}$ such that for any u -tuple (s_1, \dots, s_u) and v -tuple (t_1, \dots, t_v) with $1 \leq s_1 \leq \dots \leq s_u < t_1 \leq \dots \leq t_v \leq n$, we have

$$\left| \text{Cov} \left(\prod_{i=1}^u X_{s_i,n}, \prod_{j=1}^v X_{t_j,n} \right) \right| \leq K_n^2 M_n^{u+v} \{(u+v)!\}^{b_n} \Psi(u, v) \rho_n(t_1 - s_u), \quad (41.3)$$

where the sequence $\{\rho_n(i)\}_{i \geq 0}$ satisfies

$$\sum_{s=0}^{\infty} (s+1)^k \rho_n(s) \leq L_{1,n} L_{2,n}^k (k!)^{a_n}, \text{ for any } k \in \mathbb{N}. \quad (41.4)$$

Moreover, we require that the following moment condition holds:

$$\mathbb{E}|X_{i,n}|^k \leq (k!)^{b_n} M_n^k, \text{ } i = 1, \dots, n, \text{ for any } k \in \mathbb{N}.$$

Then, for any $n \geq 1$ and any $x > 0$, we have

$$\mathbb{P}(S_n \geq x) \leq \exp \left\{ - \frac{x^2}{C_{1,n} n + C_{2,n} x^{(2a_n+2b_n+3)/(a_n+b_n+2)}} \right\},$$

where $C_{1,n}$ and $C_{2,n}$ are constants that can be chosen to be

$$C_{1,n} = 2^{a_n+b_n+3} K_n^2 M_n^2 L_{1,n} (K_n^2 \vee 2),$$

$$C_{2,n} = 2 \{M_n L_{2,n} (K_n^2 \vee 2)\}^{1/(a_n+b_n+2)}.$$

Proof. The proof follows that of Theorem 1 in [18] with minor modifications, as listed below. Restricted to this proof, we inherit the notation in [18] and abandon the subscript n .

Equation (30) in [18] can be strengthened to

$$\mathbb{E}|Y_j| \leq 2^{k-j-1} \{(k-j+1)!\}^b K^2 M^k \rho(t_{i+1} - t_i).$$

This leads to

$$|\bar{\mathbb{E}}(X_{t_1} \cdots X_{t_k})| \leq 2^{k-1} (k!)^b K^2 M^k \rho(t_{i+1} - t_i), \quad (41.5)$$

which corresponds to Lemma 13 in [18]. Using (41.5), we obtain that

$$\begin{aligned} \left| \Gamma(X_{t_1}, \dots, X_{t_k}) \right| &\leq \sum_{v=1}^k \sum_{\cup_{p=1}^v I_p = I} N_v(I_1, \dots, I_v) 2^{k-v} (k!)^b K^{2v} M^k \\ &\leq K^2 (K^2 \vee 2)^{k-1} M^k (k!)^b \{(k-1)!\} \min_{1 \leq i < k} \rho(t_{i+1} - t_i) \\ &\leq K^2 (K^2 \vee 2)^{k-1} M^k (k!)^b \min_{1 \leq i < k} \rho(t_{i+1} - t_i). \end{aligned}$$

Thus, we have

$$\left| \Gamma_k(S_n) \right| \leq n K^2 (K^2 \vee 2)^{k-1} M^k (k!)^{b+1} \sum_{s=0}^{n-1} (s+1)^{k-2} \rho(s). \quad (41.6)$$

Equation (41.6) corresponds to Lemma 14 in [18]. The rest follows the same technique as in [18]. \square

Lastly we consider the functional dependence setting. The first result is a Rosenthal-type inequality and is from [33].

Theorem 2.5 (Theorem 1, [33]) Assume $\{X_{t,n}\}_{t \in \mathbb{Z}}$ is of dimensional $d_n = 1$ and is generated from the model (41.1) with functional dependence measures $\theta_{m,p,n}$, which is of an additional subscript n to highlight its dependence on n . Assume further that $\mathbb{E}X_{0,n} = 0$, $\mathbb{E}|X_{0,n}|^p < \infty$, and $p > 2$. Then we have, for any $n \geq 1$,

$$\begin{aligned} \|S_n\|_{L_p} &\leq n^{1/2} \left[\frac{87p}{\log p} \sum_{j=1}^n \theta_{j,2,n} + 3(p-1)^{1/2} \right. \\ &\quad \left. \sum_{j=n+1}^{\infty} \theta_{j,p,n} + \frac{29p}{\log p} \|X_{0,n}\|_{L_2} \right] + n^{1/p} \left[\frac{87p(p-1)^{1/2}}{\log p} \right. \\ &\quad \left. \sum_{j=1}^n j^{1/2-1/p} \theta_{j,p,n} + \frac{29p}{\log p} \|X_{0,n}\|_{L_p} \right]. \end{aligned}$$

The second is a Nagaev-type inequality and is also from [33].

Theorem 2.6 (Theorem 2, [33]) Assume $\{X_{t,n}\}_{t \in \mathbb{Z}}$ is of dimensional $d_n = 1$ and is generated from the model (41.1) with functional dependence measures $\theta_{m,p,n}$. Assume further that $\mathbb{E}X_{0,n} = 0$, $\mathbb{E}|X_{0,n}|^p < \infty$, and $p > 2$. Then we have the following bounds for any $n \geq 1$.

(i) Denote

$$\mu_{j,n} := (j^{p/2-1} \theta_{j,p,n}^p)^{1/(p+1)} \quad \text{and} \quad v_n := \sum_{j=1}^{\infty} \mu_{j,n} < \infty.$$

Then, for any $x > 0$,

$$\begin{aligned} \mathbb{P}(|S_n| \geq x) &\leq c_p \frac{n}{x^p} \left(v_n^{p+1} + \|X_{0,n}\|_{L_p}^p \right) \\ &\quad + 4 \sum_{j=1}^{\infty} \exp \left(-\frac{c_p \mu_{j,n}^2 x^2}{n v_n^2 \theta_{j,2,n}^2} \right) \\ &\quad + 2 \exp \left(-\frac{c_p x^2}{n \|X_{0,n}\|_{L_2}^2} \right), \end{aligned}$$

where $c_p > 0$ is a constant only depending on p .

(ii) Assume that $\Theta_{m,p,n} := \sum_{k=m}^{\infty} \theta_{k,p,n} = O(m^{-\alpha})$ as m goes to infinity, with some constant $\alpha > 1/2 - 1/p$. Then there exist absolute positive constants C_1 and C_2 such that, for any $x > 0$,

$$\mathbb{P}(|S_n| \geq x) \leq \frac{C_1 \Theta_{0,p,n}^p n}{x^p} + 4G_{1-2/p} \left(\frac{C_2 x}{\sqrt{n} \Theta_{0,p,n}} \right),$$

where for any $y > 0$, $q > 0$, $G_q(y)$ is defined to be

$$G_q(y) = \sum_{j=1}^{\infty} \exp(-j^q y^2).$$

(iii) If $\Theta_{m,p,n} = O(m^{-\alpha})$ as m goes to infinity, with some constant $\alpha < 1/2 - 1/p$, then

$$\begin{aligned} \mathbb{P}(|S_n| \geq x) &\leq \frac{C_1 \Theta_{0,p,n}^p n^{p(1/2-\alpha)}}{x^p} + 4G_{(p-2)/(p+1)} \\ &\quad \left(\frac{C_2 x}{n^{(2p-1-2\alpha p)/(2+2p)} \Theta_{0,p,n}} \right). \end{aligned}$$

It should be noted that Theorems 2.5 and 2.6 actually apply to cases beyond the sample sum, and the same inequalities hold for the partial sum process

$$S_n^* := \max_{1 \leq k \leq n} \left| \sum_{i=1}^k X_{i,n} \right|.$$

However, if S_n , instead of S_n^* , is of interest, Theorem 2.6 could be further strengthened, as was made in [44]. To this end, let's first introduce the dependence adjusted norm (DAN) for the process $\{X_{t,n}\}_{t \in \mathbb{Z}}$ as

$$\|X_{\cdot,n}\|_{p,\alpha} := \sup_{m \geq 0} (m+1)^\alpha \Theta_{m,p,n}. \quad (41.7)$$

Theorem 2.7 (Theorem 2, [44]) Assume $\{X_{t,n}\}_{t \in \mathbb{Z}}$ is of dimensional $d_n = 1$ and is generated from the model (41.1) with functional dependence measures $\theta_{m,p,n}$. Assume further that $\mathbb{E}X_{0,n} = 0$ and $\|X_{\cdot,n}\|_{p,\alpha} < \infty$ for some $p > 2$ and $\alpha > 0$. Let

$$a_n = \begin{cases} 1, & \text{if } \alpha > 1/2 - 1/p, \\ n^{p/2-1-\alpha p}, & \text{if } \alpha < 1/2 - 1/p. \end{cases}$$

Then there exist positive constants C_1, C_2 , and C_3 only depending on q and α such that, for all $x > 0$, we have

$$\mathbb{P}(|S_n| \geq x) \leq C_1 \frac{a_n n \|X_{\cdot,n}\|_{p,\alpha}^p}{x^p} + C_2 \exp \left(-\frac{C_3 x^2}{n \|X_{\cdot,n}\|_{2,\alpha}^2} \right).$$

41.2.2 Sample Sum for Random Vectors with $d_n \geq 1$

In this section we will concern the sample sum case when $d = d_n$ potentially diverges with n . Let $\{X_{t,n}\}_{t \in \mathbb{Z}}$ be a d_n -dimensional real time series of the form (41.1):

$$X_{t,n} = g_n(\dots, \epsilon_{t-1,n}, \epsilon_{t,n}) = \begin{pmatrix} g_{1,n}(\dots, \epsilon_{t-1,n}, \epsilon_{t,n}) \\ g_{2,n}(\dots, \epsilon_{t-1,n}, \epsilon_{t,n}) \\ \vdots \\ g_{d,n}(\dots, \epsilon_{t-1,n}, \epsilon_{t,n}) \end{pmatrix}. \tag{41.8}$$

Assume $\mathbb{E}X_{t,n} = 0$ and let $S_n = \sum_{i=1}^n X_{i,n}$. Theorem 2.8 below provides a tail probability for $|S_n|_\infty$, where for any vector $v = (v_1, \dots, v_d)^\top$ let $|v|_\infty = \max_{j \in [d]} |v_j|$. Assume $\mathbb{E}|X_{i,n}|^q < \infty, q > 2$, and define the uniform functional dependence measure

$$\delta_{i,q,n} = \||X_{i,n} - X_{i,\{0\},n}|_\infty\|_{L_q}, \tag{41.9}$$

where

$$|X_{i,n} - X_{i,\{0\},n}|_\infty := \max_{j \leq d} \left| g_{j,n}(\dots, \epsilon_{i-1,n}, \epsilon_{i,n}) - g_{j,n}(\dots, \epsilon_{-2,n}, \epsilon_{-1,n}, \epsilon'_0, \epsilon_{1,n}, \dots, \epsilon_{i,n}) \right|. \tag{41.10}$$

Define the vector version DAN (cf. (41.7)) as

$$\||X_{\cdot,n}|_\infty\|_{q,\alpha} = \sup_{m \geq 0} (m+1)^\alpha \Omega_{m,q,n}, \text{ where } \Omega_{m,q,n} = \sum_{i=m}^\infty \delta_{i,q,n}. \tag{41.11}$$

The constants $C_{q,\alpha} > 0$ in Theorem 2.8 only depend on q and α , and their values may change from place to place.

Theorem 2.8 (Theorem 6.2 in [45]) Assume $\||X_{\cdot,n}|_\infty\|_{q,\alpha} < \infty$, where $q > 2, \alpha > 0$. Let

$$\Psi_{2,\alpha,n} = \max_{j \leq d} \|X_{j,n}\|_{q,\alpha}$$

be the counterpart of $\||X_{\cdot,n}|_\infty\|_{q,\alpha}$ with the maximum over $j \in [d_n]$ taken outside instead of inside the expectation. Let $\ell_n = \max(1, \log d_n)$. Then the following two statements hold:

(i) If $\alpha > 1/2 - 1/q$, then for all $x \geq C_{q,\alpha}(\sqrt{n\ell_n}\Psi_{2,\alpha,n} + n^{1/q}\ell_n^{3/2}\||X_{\cdot,n}|_\infty\|_{q,\alpha})$, we have

$$\mathbb{P}\{|S_n|_\infty \geq x\} \leq C_{q,\alpha} \frac{n\ell_n^{q/2}\||X_{\cdot,n}|_\infty\|_{q,\alpha}^q}{x^q} + C_{q,\alpha} \exp\left(-C_{q,\alpha} \frac{x^2}{n\Psi_{2,\alpha,n}^2}\right).$$

(ii) If $\alpha < 1/2 - 1/q$, then for all $x \geq C_{q,\alpha}(\sqrt{n\ell_n}\Psi_{2,\alpha,n} + n^{1/2-\alpha}\ell_n^{3/2}\||X_{\cdot,n}|_\infty\|_{q,\alpha})$, we have

$$\mathbb{P}\{|S_n|_\infty \geq x\} \leq C_{q,\alpha} \frac{n^{q/2-\alpha q} \ell_n^{q/2} \||X_{\cdot,n}|_\infty\|_{q,\alpha}^q}{x^q} + C_{q,\alpha} \exp\left(-C_{q,\alpha} \frac{x^2}{n\Psi_{2,\alpha,n}^2}\right).$$

Example 2.1 As an application of Theorem 2.8, consider the following example, with the subscript n omitted for presentation simplicity. Let

$$W_i = \sum_{j=0}^\infty a_j \epsilon_{i-j}$$

be a linear process, where ϵ_j are i.i.d. innovations with finite q th moment $\mu_q := \|\epsilon_i\|_{L_q} < \infty, q > 2$, and a_j are coefficients satisfying $a_* := \sup_{m \geq 0} (m+1)^\alpha \sum_{i=m}^\infty |a_i| < \infty$. Let

$$X_{ij} = g_j(W_i) - E g_j(W_i),$$

where g_j are Lipschitz continuous functions with constants bounded by L . Then

$$|X_i - X_{i,\{0\}}|_\infty \leq L|a_i|\|\epsilon_0 - \epsilon'_0\| \text{ and } \delta_{i,q} \leq 2L|a_i|\mu_q.$$

The dependence adjusted norms

$$\|X_{\cdot,j}\|_{q,\alpha} \leq 2L\mu_q a_* \text{ and } \||X_{\cdot}|_\infty\|_{q,\alpha} \leq 2L\mu_q a_*.$$

In comparison with Theorem 2.7, the bound in Theorem 2.8 is sharp up to a multiplicative logarithmic factor $(\log d)^{q/2}$, adjusting for multi-dimensionality.

Example 2.2 (Largest Eigenvalues of Sample Auto-Covariance Matrices) Let W_i in Example 2.1 be of the form of stationary causal process (41.1) with $\mathbb{E}W_i = 0, \mathbb{E}|W_i|^q < \infty, q > 2$. Again let's omit the subscript n for no confusion will be made. Let

$$a_i = \|W_i - W'_i\|_{L_q}$$

be the associated functional dependence measure, and assume the dependence adjusted norm $\|X_{\cdot}\|_{q,\alpha} < \infty, \alpha > 1/2 - 1/q$. Let

$$S_n(\theta) = \sum_{t=1}^n W_t \exp(\sqrt{-1}t\theta), \quad 0 \leq \theta \leq 2\pi,$$

be the Fourier transform of $(W_t)_{t=1}^n$, where $\sqrt{-1}$ is the imaginary unit. Let

$$d = n^9 \text{ and } \lfloor \theta \rfloor_d = 2\pi \lfloor d\theta / (2\pi) \rfloor / d.$$

By Theorem 2.8(i), the inequality therein holds with

$$\max_{0 \leq \theta \leq 2\pi} |S_n(\lfloor \theta \rfloor_d)| = \max_{j \leq d} |S_n(2\pi j/d)|.$$

Noting that

$$\| \max_{\theta} |S_n(\lfloor \theta \rfloor_d) - S_n(\theta)| \|_q \leq \|W_1\|_q / n^6.$$

Thus with elementary manipulations the same inequality in Theorem 2.8(i) holds with the term $\max_{0 \leq \theta \leq 2\pi} |S_n(\theta)|$.

Given $(W_t)_{t=1}^n$, let the sample covariance matrix

$$\widehat{\Sigma}_n = (\widehat{\gamma}_{j-k}), \text{ where } \widehat{\gamma}_k = n^{-1} \sum_{l=k+1}^n W_l W_{l-k}, \quad 0 \leq k \leq n-1,$$

Notice that the largest eigenvalues

$$\lambda_{\max}(\widehat{\Sigma}_n) \leq \max_{0 \leq \theta \leq 2\pi} |S_n(\theta)|^2 / n.$$

We obtain the tail probability inequality

$$\begin{aligned} \mathbb{P}(\lambda_{\max}(\widehat{\Sigma}_n) \geq u) &\leq \mathbb{P}\left(\max_{0 \leq \theta \leq 2\pi} |S_n(\theta)| \geq (nu)^{1/2}\right) \\ &\leq C_{q,\alpha} \frac{n(\log n)^{q/2} \|X\|_{q,\alpha}^q}{(nu)^{q/2}} \\ &\quad + C_{q,\alpha} \exp\left(-C_{q,\alpha} \frac{u}{\|X\|_{2,\alpha}^2}\right), \end{aligned}$$

when $u \geq C_{q,\alpha} \|X\|_{2,\alpha}^2 \log n$ for a sufficient large constant $C_{q,\alpha}$.

41.2.3 Sample Sum for Random Matrices with $d_n \geq 1$

In this section we will consider the case of time-dependent random matrices. Here $X_{t,n} \in \mathbb{R}^{d_n \times d_n}$ is a d_n -dimensional random matrix, and $\{X_{t,n}\}_{t \in \mathbb{Z}}$ is a matrix-valued time series. Tail probability inequalities for spectral norms for the sum $\sum_{t=1}^n X_{t,n}$ will be presented. The latter results are useful for statistical inference of auto-covariance matrices generated from high-dimensional time series.

Ahlswede and Winter [2], Oliveira [37], and Tropp [41], among many others, have studied such bounds when $\{X_{t,n}\}_{t \in \mathbb{Z}}$ are mutually independent. For instance, [37] and [41] have introduced the following Bernstein-type inequality for tails.

The result in [37] also applies to martingales (cf. Freedman’s Inequality for matrix martingales [19]). Also see [34] for further extensions to conditionally independent sequences and combinatorial sums.

Theorem 2.9 (Corollary 7.1 in [37], Theorem 1.4 in [41])

Let $X_{1,n}, \dots, X_{n,n}$ be real, mean-zero, symmetric independent $d_n \times d_n$ random matrices and assume there exists a positive constant M_n such that $\lambda_{\max}(X_{i,n}) \leq M_n$ for all $1 \leq i \leq n$. Then for any $x \geq 0$,

$$\mathbb{P}\left\{\lambda_{\max}\left(\sum_{i=1}^n X_{i,n}\right) \geq x\right\} \leq d_n \exp\left(-\frac{x^2}{2\sigma_n^2 + 2M_n x/3}\right),$$

where $\sigma_n^2 := \lambda_{\max}(\sum_{i=1}^n \mathbb{E}X_{i,n}^2)$ and recall that $\lambda_{\max}(\cdot)$ represents the largest eigenvalue of the input.

Assuming $\{X_{t,n}\}_{t \in \mathbb{Z}}$ satisfies a geometrically β -mixing decaying rate:

$$\beta(\{X_{t,n}\}_{t \in \mathbb{Z}}; m) \leq \exp\{-\gamma_n(m-1)\}, \quad \text{for } m = 1, 2, \dots \tag{41.12}$$

with some constant $\gamma_n > 0$ possibly depending on n , [6] proved the following theorem that extends the matrix Bernstein inequality to the β -mixing case. In the sequel, for any set A , we denote $\text{Card}(A)$ to be its cardinality.

Theorem 2.10 (Theorem 1 in [6]) Let $\{X_{t,n}\}_{t \in \mathbb{Z}}$ be a sequence of mean-zero symmetric $d_n \times d_n$ random matrices with $\sup_{i \in [n]} \lambda_{\max}(X_{i,n}) \leq M_n$ for some positive constant M_n . Further assume the β -mixing condition (41.12) holds. Then there exists a universal positive constant C such that, for any $n \geq 2$ and $x > 0$,

$$\begin{aligned} \mathbb{P}\left\{\lambda_{\max}\left(\sum_{i=1}^n X_{i,n}\right) \geq x\right\} \\ \leq d_n \exp\left\{-\frac{Cx^2}{v_n^2 n + M_n^2/\gamma_n + xM_n \tilde{\gamma}(\gamma_n, n)}\right\}, \end{aligned}$$

where

$$\begin{aligned} v^2 &:= \sup_{K \subset [n]} \frac{1}{\text{Card}(K)} \lambda_{\max}\left\{\mathbb{E}\left(\sum_{i \in K} X_{i,n}\right)^2\right\} \quad \text{and} \\ \tilde{\gamma}(\gamma_n, n) &:= \frac{\log n}{\log 2} \max\left(2, \frac{32 \log n}{\gamma_n \log 2}\right). \end{aligned}$$

Later, this result is further extended to the τ -mixing case, which was made in [23].

Theorem 2.11 (Theorem 4.3 in [23]) Consider a sequence of real, mean-zero, symmetric $d_n \times d_n$ random matrices

$\{X_{t,n}\}_{t \in \mathbb{Z}}$ with $\sup_{i \in [n]} \|X_{i,n}\| \leq M_n$ for some positive constant M_n that is allowed to depend on n and $\|\cdot\|$ represents the matrix spectral norm. In addition, assume that this sequence is of a geometrically decaying τ -mixing rate, i.e.,

$$\tau(\{X_{t,n}\}_{t \in \mathbb{Z}}; m, \|\cdot\|) \leq M_n \psi_{1,n} \exp\{-\psi_{2,n}(m-1)\},$$

for $m = 1, 2, \dots$

with some constants $\psi_{1,n}, \psi_{2,n} > 0$. Denote $\tilde{\psi}_{1,n} := \max\{d_n^{-1}, \psi_{1,n}\}$. Then for any $x \geq 0$ and any $n \geq 2$, we have

$$\mathbb{P}\left\{\lambda_{\max}\left(\sum_{i=1}^n X_{i,n}\right) \geq x\right\} \leq d_n \exp\left\{-\frac{x^2}{8(15^2 n v_n^2 + 60^2 M_n^2 / \psi_{2,n}) + 2x M_n \tilde{\psi}(\tilde{\psi}_{1,n}, \psi_{2,n}, n, d_n)}\right\},$$

where

$$v_n^2 := \sup_{K \subset [n]} \frac{1}{\text{Card}(K)} \lambda_{\max}\left\{\mathbb{E}\left(\sum_{i \in K} X_{i,n}\right)^2\right\}$$

and $\tilde{\psi}(\tilde{\psi}_{1,n}, \psi_{2,n}, n, d_n) := \frac{\log n}{\log 2} \max\left\{1, \frac{8 \log(\tilde{\psi}_{1,n} n^6 d_n)}{\psi_{2,n}}\right\}$.

We note that the above matrix Bernstein inequalities for weakly dependent data can be immediately applied to study the behavior of many statistics of importance in analyzing a high-dimensional time series model. In particular, tail behaviors for the largest eigenvalues of sample auto-covariances in weakly dependent high-dimensional time series models have been characterized in Theorems 2.1 and 2.2 in [23], with bounds delivered for both general and Gaussian weakly dependent time series (the later using a different set of techniques tailored for Gaussian processes) separately; see also Sect. 41.4 ahead for a concrete example.

41.2.4 U- and V-Statistics

Consider $\{X_{i,n}\}_{i \in [n]}$ to be n random variables of identical distribution in a measurable space $(\mathcal{X}, \mathcal{B}_{\mathcal{X}})$. Given a symmetric kernel function $h_n(\cdot) : \mathcal{X}^r \rightarrow \mathbb{R}$, the U- and V-statistic $U_n(h_n)$ and $V_n(h_n)$ of order r_n are defined as

$$U_n(h_n) := \binom{n}{r_n}^{-1} \sum_{1 \leq i_1 < \dots < i_{r_n} \leq n} h_n(X_{i_1,n}, \dots, X_{i_{r_n},n})$$

and $V_n(h_n) := n^{-r_n} \sum_{i_1, \dots, i_{r_n}=1}^n h_n(X_{i_1,n}, \dots, X_{i_{r_n},n})$.

The V- and U-statistics are popular alternatives to sample sums and have been routinely used in statistics nowadays (cf. the textbooks [31] and [28]).

Nonasymptotic probability and moment inequalities for V- and U-statistics in the i.i.d. case have been extensively studied [1, 3, 20, 24]. Assuming $\{X_{t,n}\}_{t \in \mathbb{Z}}$ to be geometrically ϕ -mixing, [22] established the following theorem that gives an exponential inequality for dependent U-statistics.

Theorem 2.12 (Theorem 2.1, [22]) *Let $\{X_{t,n}\}_{t \in \mathbb{Z}}$ satisfy*

$$\phi(\{X_{t,n}\}_{t \in \mathbb{Z}}; m) \leq c_n \exp(-C_n m) \text{ for } m = 1, 2, \dots$$

with two constants $c_n, C_n > 0$. Assume further that

$$\|h_n\|_{\infty} \leq M_n,$$

symmetric and is mean-zero (i.e., $\mathbb{E}h_n = 0$ with regard to the product measure). Then there exist two constants $c'_n, C'_n > 0$ that only depend on c_n, C_n , and r_n , such that, for any $x \geq 0$ and $n \geq 4$,

$$\begin{aligned} \mathbb{P}(|U_n(h_n)| \geq c'_n M_n / \sqrt{n} + x) \\ \leq 2 \exp\left(-\frac{C'_n x^2 n}{M_n^2 + M_n x (\log n) (\log \log 4n)}\right). \end{aligned}$$

With tedious calculations, the dependence of c'_n, C'_n on r_n, c_n, C_n in Theorem 2.12 can be explicitly obtained, as was made in Theorems 2.10 and 2.11.

In order to present the next result, let's first introduce more concepts in U- and V-statistics. For presentation clearness, let us assume the kernel $h_n(\cdot)$, its order r_n , and the dimension d_n are fixed and hence written as $h(\cdot)$, r , and d without the subscript. Assume $\{X_{t,n}\}_{t \in \mathbb{Z}}$ to be stationary for any $n \in \mathbb{N}$. Let $\{\tilde{X}_{i,n}\}_{i \in [n]}$ be an i.i.d. sequence with $\tilde{X}_{1,n}$ identically distributed as $X_{1,n}$. The mean value of a symmetric kernel h (with regard to the marginal probability measure \mathbb{P}_n) is defined as

$$\theta_n := \theta_n(h) := \mathbb{E}h(\tilde{X}_{1,n}, \dots, \tilde{X}_{r,n}).$$

The kernel $h(\cdot)$ is called degenerate of level $k-1$ ($2 \leq k \leq r$) with regard to the measure \mathbb{P}_n if and only if

$$\mathbb{E}h(x_1, \dots, x_{k-1}, \tilde{X}_{k,n}, \dots, \tilde{X}_{r,n}) = \theta_n$$

for any $(x_1^\top, \dots, x_{k-1}^\top)^\top \in \text{supp}(\mathbb{P}_n^{k-1})$, the support of the product measure \mathbb{P}_n^{k-1} .

When h is degenerate of level $k - 1$, its Hoeffding decomposition takes the form

$$h(x_1, \dots, x_r) - \theta_n = \sum_{1 \leq i_1 < \dots < i_k \leq r} h_{k,n}(x_{i_1}, \dots, x_{i_k}) + \dots + h_{r,n}(x_1, \dots, x_r),$$

where $\{h_{p,n}\}_{p=k}^r$ are recursively defined as

$$h_{1,n}(x) := g_{1,n}(x),$$

$$h_{p,n}(x_1, \dots, x_p) := g_{p,n}(x_1, \dots, x_p) - \sum_{k=1}^p h_{1,n}(x_k) - \dots - \sum_{1 \leq k_1 < \dots < k_{p-1} \leq p} h_{p-1,n}(x_{k_1}, \dots, x_{k_{p-1}}),$$

for $p = 2, \dots, r$, with $\{g_{p,n}\}_{p=1}^r$ defined as $g_{r,n} := h - \theta_n$, and

$$g_{p,n}(x_1, \dots, x_p) := \mathbb{E}h(x_1, \dots, x_p, \tilde{X}_{p+1,n}, \dots, \tilde{X}_{r,n}) - \theta_n$$

for $1 \leq p \leq r - 1$. For each $1 \leq p \leq r$, we denote the V-statistic generated by $h_{p,n}$ by

$$V_n(h_{p,n}) := n^{-p} \sum_{i_1, \dots, i_p=1}^n h_{p,n}(X_{i_1,n}, \dots, X_{i_p,n}).$$

Theorem 2.13 (A Slight Modification to Theorem 1 in [40]) *Suppose $\{X_{i,n}\}_{i=1}^n$ is part of a stationary sequence $\{X_{t,n}\}_{t \in \mathbb{Z}}$ that is geometrically α -mixing with coefficient*

$$\alpha(\{X_{t,n}\}_{t \in \mathbb{Z}}; m) \leq c_n \exp(-C_n m) \quad \text{for all } m \geq 1,$$

where c_n and C_n are two positive constants. Suppose $h \in L_1(\mathbb{R}^{rd})$ is fixed, symmetric, continuous, and its Fourier transform $\widehat{h}(u) := \int h(x)e^{-2\pi i u^\top x} dx$ satisfies

$$\int_{\mathbb{R}^{rd}} |\widehat{h}(u)| \|u\|^q du < \infty$$

for some $q \geq 1$, where $\|\cdot\|$ represents the Euclidean norm. Then, there exists a positive constant $C'_n = C(r, c_n, C_n)$ such that for each $1 \leq p \leq r$, and any $n \geq 2, x > 0$,

$$\mathbb{P}\left(|V_n(h_{p,n})| \geq x\right) \leq 6 \exp\left\{-\frac{C'_n n x^{2/p}}{A_{p,n}^{1/p} + x^{1/p} M_{p,n}^{1/p}}\right\}$$

with

$$A_{p,n} = 2^{2r} \|\widehat{h}\|_{L_1}^2 \left\{ \frac{64c_n^{1/3}}{1 - \exp(-C_n/3)} + \frac{(\log n)^4}{n} \right\}^p \quad \text{and}$$

$$M_{p,n} = 2^r \|\widehat{h}\|_{L_1} (\log n)^{2p}.$$

As Theorem 2.12, the dependence of C'_n on r, c_n, C_n in the above theorem could be explicitly calculated. We also note that, though $h(\cdot)$ itself is assumed to be fixed, the ‘‘degenerate’’ kernels $h_{p,n}$ could depend on n through the measure \mathbb{P}_n in the triangular array setting, and hence the subscript n is kept.

41.3 A Cautionary Example

Section 41.2 exemplifies the use of dependence measures to construct desired moment/probability inequalities for quantifying the statistical properties of procedures in a high-dimensional time series model. The problem then reduces to characterizing these dependence measures in a triangular array setting as highlighted at the beginning of Sect. 41.2. As is apparent from reading their definitions, those dependence measures introduced in [13, 17], and [42] can be explicitly calculated. Therefore, the verification of those dependence measures, as were made in Section 3 in [15] as well as in [23, 42], and many other places, are obviously still valid under the high-dimensional triangular array framework.

The verifications for the mixing conditions introduced at the beginning of Sect. 41.1, on the other hand, should be checked with caution under this new framework. In the following we will use the example of β -mixing to showcase this new challenge of high dimensionality in establishing mixing-type dependence for time series data.

In literature, for a time series model that is fixed (i.e., not changing when more data points are observed), there have been a variety of results to establish bounds for β -mixing coefficients. See, for example, [32] for a review and [11] for β -mixing of Markov processes. Let’s focus on a particular example. Consider the following simple d -dimensional stationary Gaussian VAR(1) model:

$$X_t = \kappa X_{t-1} + E_t = \sum_{j=0}^{\infty} \kappa^j E_{t-j}, \quad \text{for all } t \in \mathbb{Z}. \quad (41.13)$$

Here the autocorrelation coefficient $\kappa \in \mathbb{R}$ is assumed to be fixed and satisfy $0 < \kappa < 1$ for simplicity, and the innovation noises $\{E_t \in \mathbb{R}^d\}$ are i.i.d. Gaussian. Then it is immediate (cf. Proposition 2 in [32]) that $\{X_t\}$ is geometrically β -mixing satisfying

$$\beta(\{X_t\}; m) \leq C\gamma^m \quad (41.14)$$

for some fixed constants $C > 0, \gamma < 1$.

However, in high dimensions such a derivation is problematic. Let’s fix the framework first. Adopting the triangular array setting as described in the last section, we assume that the studied model could change as more observations are

available to us. In other words, let's adopt a parallel model to Eq. (41.13): for any $n = 1, 2, 3, \dots$, write

$$X_{t,n} = \begin{pmatrix} X_{t,1,n} \\ X_{t,2,n} \\ \vdots \\ X_{t,d_n,n} \end{pmatrix} = \begin{pmatrix} \kappa_n & 0 & \dots & 0 \\ 0 & \kappa_n & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \kappa_n \end{pmatrix} \underbrace{\begin{pmatrix} X_{t-1,1,n} \\ X_{t-1,2,n} \\ \vdots \\ X_{t-1,d_n,n} \end{pmatrix}}_{X_{t-1,n} \in \mathbb{R}^{d_n}} + \underbrace{\begin{pmatrix} E_{t,1,n} \\ E_{t,2,n} \\ \vdots \\ E_{t,d_n,n} \end{pmatrix}}_{E_{t,n} \in \mathbb{R}^{d_n}},$$

for all $t \in \mathbb{Z}$.
(41.15)

Here for any n , the observed data $\{X_{1,n}, X_{2,n}, \dots, X_{n,n}\}$ are assumed to be generated from a process $\sum_{j=0}^{\infty} \kappa_n^j E_{t-j,n}$, where first of all the dimension of the time series d_n has been allowed to change with the sample size n . Moreover, as an implicit consequence of the above high-dimensional triangular array framework, all the parameters in Model (41.15), including $\kappa_n \in \mathbb{R}$, $\text{Cov}(X_{t,n}) \in \mathbb{R}^{d_n \times d_n}$, and $\text{Cov}(E_{t,n}) \in \mathbb{R}^{d_n \times d_n}$, are now allowed to change as the sample size n is increasing.

Once such a framework is fixed, it becomes clear that the analysis of various dependence conditions has to be nonasymptotic, i.e., we now have to provide an analysis of the β -mixing coefficient that takes the change of d_n, κ_n , and all the other model parameters into account. With these concepts in mind, we first state a somewhat comforting result that certain desirable properties could still be established for α -mixing (in contrast to the β -mixing) coefficient under the triangular array setting.

Theorem 3.1 Consider the following simple stationary Gaussian vector autoregressive model that generalizes (41.15) by relaxing restrictions on the transition matrix:

$$X_{t,n} = A_n X_{t-1,n} + E_{t,n}, \quad t \in \mathbb{Z}. \quad (41.16)$$

We then have

$$\alpha(\{X_{t,n}\}_{t \in \mathbb{Z}}; m) \leq \left\{ \frac{\lambda_{\max}(\Sigma_n)}{\lambda_{\min}(\Sigma_n)} \right\}^{1/2} \|A_n\|^m,$$

where $\Sigma_n := \text{Cov}(X_{0,n})$, $\lambda_{\min}(\cdot)$ stands for the smallest eigenvalue of the input, and $\|\cdot\|$ is the matrix spectral norm.

Proof. For notation simplicity, let's remove n from the subscript. Since VAR(1) is a stationary Markov chain, by [9], we have the ρ -mixing coefficient

$$\rho\{\sigma(X_{-\infty}^0), \sigma(X_m^\infty)\} = \rho\{\sigma(X_0), \sigma(X_m)\}.$$

By Theorem 1 from [27], if $U_1, U_2, \dots, U_m, V_1, V_2, \dots, V_\ell$ are jointly normal random variables, then there exist real numbers $a_1, a_2, \dots, a_m, b_1, b_2, \dots, b_\ell$ such that

$$\begin{aligned} & \rho\{\sigma(U_k, 1 \leq k \leq m), \sigma(V_k, 1 \leq k \leq \ell)\} \\ &= \text{Corr}\left(\sum_{k=1}^m a_k U_k, \sum_{k=1}^{\ell} b_k V_k\right). \end{aligned}$$

Since (X_0, X_m) is multivariate normal, there exist real numbers $a = (a_1, a_2, \dots, a_p)^\top, b = (b_1, b_2, \dots, b_p)^\top$ such that

$$\begin{aligned} \rho\{\sigma(X_0), \sigma(X_m)\} &= \text{Corr}(a^\top X_0, b^\top X_m) = \frac{a^\top \Sigma(A^m)^\top b}{\sqrt{a^\top \Sigma a b^\top \Sigma b}} \\ &\leq \sqrt{\|\Sigma^{\frac{1}{2}}(A^m)^\top \Sigma^{-1}(A^m) \Sigma^{\frac{1}{2}}\|}, \end{aligned}$$

where the last inequality is followed by Cauchy-Schwarz. Hence we have

$$\rho\{\sigma(X_{-\infty}^0), \sigma(X_m^\infty)\} \leq \left\{ \frac{\lambda_{\max}(\Sigma)}{\lambda_{\min}(\Sigma)} \right\}^{\frac{1}{2}} \|A\|^m.$$

Now noticing

$$\alpha\{\sigma(X_{-\infty}^0), \sigma(X_m^\infty)\} \leq \rho\{\sigma(X_{-\infty}^0), \sigma(X_m^\infty)\}$$

finishes the proof. □

Applying Theorem 3.1 to Model (41.15), it is clear that the α -mixing coefficient for Model (41.15) is bounded by κ_n^m , which will be exponentially tending to 0 if $\kappa_n < 1$ is fixed, regardless of how large the dimension d_n is. We then state a possibly striking result, that, even if restricting to the Model (41.15) and fixing κ_n , the β -mixing coefficient of the time series $\{X_{t,n}\}_{t \in \mathbb{Z}}$ could still be tending to 1 if d_n is sufficiently larger than the time gap. Thusly, β -mixing coefficient is dimension dependent.

Theorem 3.2 Consider the model (41.15) under the triangular array setting. If we further assume that $E_{0,n} = (E_{0,1,n}, \dots, E_{0,d_n,n})^\top$ have i.i.d. components, then for any positive integers n and m , we have

$$\beta(\{X_{t,n}\}_{t \in \mathbb{Z}}; m) \geq 1 - 2 \exp\left(-\frac{d_n \kappa_n^{2m}}{18\pi^2}\right).$$

In particular, if (1) $d_n = d$ is not changing with n but $\lim_{n \rightarrow \infty} \kappa_n^{2n} > \frac{18\pi^2 \log 2}{d}$, or (2) $\kappa_n = \kappa$ is not changing with n but $\lim_{n \rightarrow \infty} d_n \kappa^{2n} > 18\pi^2 \log 2$, then $\liminf_{n \rightarrow \infty} \beta(\{X_{t,n}\}_{t \in \mathbb{Z}}; n) > 0$.

Theorem 3.2 is concerning a particularly simple model that is merely aggregating d_n i.i.d. AR(1) Gaussian sequences once we have n data points. It is very unlikely that any assumption in a general theorem for quantifying the behavior of a high-dimensional time series could exclude such a simple case. However, it has been apparent from this result that, once

the triangular array framework is adopted, many simple and elegant properties like Eq. (41.14) could no longer be trusted because otherwise, the case that $\lim_{n \rightarrow \infty} \beta(\{X_{t,n}\}_{t \in \mathbb{Z}}; n) \neq 0$ shall never happen. The reason is, once the model $\{X_{t,n}\}$ is allowed to change with n , the values of C and γ in (41.14) will depend on the sample size n . Any solid analysis of the β -mixing coefficient hence has to be fully nonasymptotic. This, however, violates the spirit beneath the definition of various mixing coefficients and, to the authors' knowledge, cannot be trivially handled (except for the α - and ρ -mixing coefficients under a Gaussian process, as showcased above in Theorem 3.1).

Proof of Theorem 3.2 The proof is nonasymptotic and relies on several known results in the mixing literature. For presentation clearness, we omit the subscript n in the following when no confusion is made.

In the first step, we need to establish a lower bound for the marginal α -mixing coefficient $\alpha(\sigma(X_{0,1}), \sigma(X_{m,1}))$ with the understanding that

$$X_t = (X_{t,1}, \dots, X_{t,d})^\top.$$

For any bivariate Gaussian random vector $(Z_1, Z_2)^\top \in \mathbb{R}^2$, the following two facts are known.

(1) One has

$$\begin{aligned} \alpha(\sigma(Z_1), \sigma(Z_2)) &\leq \rho(\sigma(Z_1), \sigma(Z_2)) \\ &\leq 2\pi\alpha(\sigma(Z_1), \sigma(Z_2)). \end{aligned}$$

See, for example, Equation (1.9) in Chapter 4 of [25] or Theorem 2 in [27].

(2) Theorem 1 in [27] gives

$$\rho(\sigma(Z_1), \sigma(Z_2)) = |\text{Corr}(Z_1, Z_2)|.$$

The above two results then yield

$$\alpha(\sigma(X_{0,1}), \sigma(X_{m,1})) \geq \frac{1}{2\pi} |\text{Corr}(X_{0,1}, X_{0,m})| = \frac{\kappa^m}{2\pi}.$$

In the second step, we are going to establish a lower bound on $\beta(\sigma(X_0), \sigma(X_m))$ based on the derived lower bound for the marginal α -mixing coefficient. For any $j \in [d]$, since

$$\alpha(\sigma(X_{0,j}), \sigma(X_{m,j})) \geq \frac{\kappa^m}{2\pi},$$

by definition, there must exist sets $G \in \sigma(X_{0,j})$ and $H \in \sigma(X_{m,j})$ such that

$$\begin{aligned} \left| \mathbb{P}(X_{0,j} \in G, X_{m,j} \in H) - \mathbb{P}(X_{0,j} \in G)\mathbb{P}(X_{m,j} \in H) \right| \\ \geq \frac{\kappa^m}{3\pi} =: \eta. \end{aligned}$$

Without loss of generality, we may assume that

$$\begin{aligned} \theta := \mathbb{P}(X_{0,j} \in G, X_{m,j} \in H) &> \mathbb{P}(X_{0,j} \in G)\mathbb{P}(X_{m,j} \in H) \\ &=: \xi. \end{aligned}$$

For $j \in [d]$, let's define V_j, W_j as

$$V_j = \mathbf{1}(X_{0,j} \in G) \quad \text{and} \quad W_j = \mathbf{1}(X_{m,j} \in H),$$

where $\mathbf{1}(\cdot)$ represents the indicator function. Then we have

- (1) For each $j \in [d]$, $\mathbb{E}V_j = \mathbb{P}(X_{0,j} \in G)$, $\mathbb{E}W_j = \mathbb{P}(X_{m,j} \in H)$, $\mathbb{E}(V_j W_j) = \theta$.
- (2) By i.i.d.-ness of $E_{0,1}, \dots, E_{0,d}$, $\{(V_j, W_j), j \in [d]\}$ is an i.i.d. sequence.

Let's now consider the following event

$$\left\{ \frac{1}{d} \sum_{j=1}^d V_j W_j \geq \theta - \frac{\eta}{2} \right\}. \tag{41.17}$$

By Hoeffding's inequality for i.i.d. data [24], we have

$$\mathbb{P}\left(\frac{1}{d} \sum_{j=1}^d V_j W_j \geq \theta - \frac{\eta}{2}\right) \geq 1 - \exp(-d\eta^2/2).$$

On the other hand, consider a comparable event to (41.17) under the product measure:

$$\left\{ \frac{1}{d} \sum_{j=1}^d V_j \tilde{W}_j \geq \theta - \frac{\eta}{2} \right\},$$

where $\{\tilde{W}_j\}_{j \in [d]}$ is a copy of $\{W_j\}_{j \in [d]}$ and is independent of $\{V_j\}_{j \in [d]}$. Again, by Hoeffding's inequality, we have

$$\begin{aligned} \mathbb{P}\left(\frac{1}{d} \sum_{j=1}^d V_j \tilde{W}_j \geq \theta - \frac{\eta}{2}\right) &\leq \mathbb{P}\left(\frac{1}{d} \sum_{j=1}^d V_j \tilde{W}_j \geq \xi + \frac{\eta}{2}\right) \\ &\leq \exp(-d\eta^2/2) \end{aligned}$$

as $\theta - \xi \geq \eta$.

By definition of β -mixing coefficient, we then have

$$\begin{aligned} \beta(\sigma(X_0), \sigma(X_m)) &\geq \left| \mathbb{P}\left(\frac{1}{d} \sum_{j=1}^d V_j W_j \geq \theta - \frac{\eta}{2}\right) \right. \\ &\quad \left. - \mathbb{P}\left(\frac{1}{d} \sum_{j=1}^d V_j \tilde{W}_j \geq \theta - \frac{\eta}{2}\right) \right| \\ &\geq 1 - 2 \exp\left(-\frac{d\kappa^{2m}}{18\pi^2}\right). \end{aligned}$$

Lastly, by noticing that the model studied is naturally a Markov chain and by using Theorem 7.3 in [10], one obtains

$$\beta(\{X_t\}; m) = \beta(\sigma(X_0), \sigma(X_m)),$$

which finishes the proof. \square

41.4 Statistical Applications

In this section, let's consider one particular example of statistical application to exemplify a (correct) use of the probability inequalities for analyzing a high-dimensional time series analysis problem. In the following we omit the subscript n when no confusion is possible.

We focus on the following simple lag-one vector autoregressive model that has been discussed in Theorem 3.1,

$$X_t = AX_{t-1} + E_t, \quad t \in \mathbb{Z}, \quad X_t \in \mathbb{R}^d. \quad (41.18)$$

In this section our task is to, given n observations X_1, \dots, X_n of a triangular array Model (41.18), estimate the transition matrix A in Model (41.18) under a low-rank assumption. Here the assumption, $r := \text{rank}(A) < d$, is regular in the related literature [4,5,7,12,29,30,36,38]. In particular, it is motivated by latent factor models, where a few latent factors drive the main movement of the multivariate time series.

The problem of transition matrix estimation is naturally related to multiple regression. For this, under the low-rank assumption on A , the least squares estimator (LSE),

$$\hat{A}^{\text{LSE}} := \operatorname{argmin}_{Q \in \mathbb{R}^{d \times d}} \frac{1}{n-1} \sum_{t=2}^n \|X_t - QX_{t-1}\|_F^2,$$

where $\|\cdot\|_F$ represents the matrix Hilbert-Schmidt norm, is not statistically efficient for estimating A [26]. For improving estimation efficiency, there have been a number of methods introduced in the literature. In particular, [36] proposed the following penalized-LSE:

$$\hat{A}_\lambda := \operatorname{argmin}_{Q \in \mathbb{R}^{d \times d}} \frac{1}{n-1} \sum_{t=2}^n \|X_t - QX_{t-1}\|_F^2 + \lambda \|Q\|_*. \quad (41.19)$$

Here the matrix nuclear norm $\|\cdot\|_*$ is added to induce the sparsity of the estimator's singular values and hence encourages low-rankness. The obtained estimator is easy to implement and proves to enjoy good empirical performance as well as preliminary theoretical properties under an additional Gaussian process assumption.

In this section we plan to give a new analysis of the estimator \hat{A}_λ based on the newly developed probability bounds in [23] (elaborated in Theorem 2.11), which is shown to be able to eliminate the Gaussian assumption that is essential in the original analysis of [36].

We start from the following general setting. Suppose $M \in \mathbb{R}^{d \times d}$ and $M_1 \in \mathbb{R}^{d \times d}$ both to be of dimension d , are two real matrices, M is symmetric and positive definite, and the matrix of interest A could be written as $A = M_1^\top M^{-1} \in \mathbb{R}^{d \times d}$ of rank $r \leq d$. Let \hat{M} and \hat{M}_1 be estimates of M and M_1 , and define

$$\hat{A}_\lambda^{\text{G}} := \operatorname{argmin}_{Q \in \mathbb{R}^{d \times d}} L_\lambda(Q; \hat{M}, \hat{M}_1),$$

$$\text{where } L_\lambda(Q; \hat{M}, \hat{M}_1) := \langle -2\hat{M}_1 + \hat{M}Q^\top, Q^\top \rangle + \lambda \|Q\|_*.$$

It is direct to check that \hat{A}_λ is a special case of $\hat{A}_\lambda^{\text{G}}$ with \hat{M} and \hat{M}_1 chosen to the sample covariance and lag-one auto-covariance matrices; this is the famous Yule-Walker formula.

The following lemma relates the analysis of \hat{A}_λ to that of analyzing the marginal and first-order auto-covariance matrices. Its proof is straightforward given the literature.

Lemma 4.1 *Assume \hat{M} and \hat{M}_1 are the estimates of $M \in \mathbb{R}^{d \times d}$ and $M_1 \in \mathbb{R}^{d \times d}$ based on n observations, satisfying*

$$\mathbb{P}(\|\hat{M} - M\| \leq \delta_1) \geq 1 - \epsilon_1 \quad \text{and} \quad \mathbb{P}(\|\hat{M}_1 - M_1\| \leq \delta_2) \geq 1 - \epsilon_2.$$

Here $\delta_1, \delta_2, \epsilon_1, \epsilon_2$ are functions of (n, d) , and ϵ_1, ϵ_2 go to zero as (n, d) increases to infinity. Further assume

$$\begin{aligned} M &= M^\top, \quad \lambda_{\min}(M) \geq \gamma_{\min}, \quad A := M_1^\top M^{-1}, \\ \text{rank}(A) &\leq r, \quad \|A\|_2 \leq \gamma_{\max}, \end{aligned}$$

and

$$\lambda \geq 2(\gamma_{\max} \delta_1 + \delta_2) \quad \text{and} \quad \mu \leq \gamma_{\min} - \delta_1,$$

where γ_{\min} and γ_{\max} are two absolute positive constants. We then have

$$\mathbb{P}\left(\|\widehat{A}_\lambda^G - A\|_F \geq \frac{\lambda + 2\sqrt{2}(\gamma_{\max}\delta_1 + \delta_2)}{2\mu} \sqrt{r}\right) \leq \epsilon_1 + \epsilon_2.$$

The next result establishes convergence of sample auto-covariance matrices in the following general VAR(p) model:

$$Y_t = A_1 Y_{t-1} + \dots + A_2 Y_{t-p} + E_t,$$

where $\{E_t\}_{t \in \mathbb{Z}}$ is a sequence of zero-mean and subgaussian independent (not necessarily identically distributed) vectors such that for all $t \in \mathbb{Z}$ and $u \in \mathbb{R}^d$, $\|u^\top E_t\|_{\psi_2} < \infty$ for some universal constant $c' > 0$. Here $\|\cdot\|_{\psi_2}$ is the Orlicz ψ_2 norm defined as

$$\|X\|_{\psi_2} := \inf \left\{ k \in (0, \infty) : \mathbb{E}[\exp\{|X|/k\}^2] - 1 \leq 1 \right\}.$$

In addition, assume $\|A_k\| \leq a_k < 1$ for all $1 \leq k \leq p$, and $\sum_{k=1}^p a_k < 1$, where $\{a_k\}_{k=1}^p$ and p are some universal constants.

Under these conditions, we have the following theorem, which is a direct implementation of Theorem 2.11.

Theorem 4.1 For the above $\{Y_t\}_{t \in \mathbb{Z}}$, let $\widehat{\Sigma}_m := (n - m)^{-1} \sum_{t=1}^{n-m} Y_t Y_t^\top$ be the m -th order auto-covariance matrix. Define

$$\begin{aligned} \kappa_1 &:= \sup_{t \in \mathbb{Z}} \sup_{u \in \mathbb{S}^{d-1}} \|u^\top Y_t\|_{\psi_2} < \infty, \\ \kappa_* &:= \sup_{t \in \mathbb{Z}} \sup_{v \in \mathbb{S}^{d-1}} \|v^\top Y_t\|_{\psi_2}, \text{ and } r^* = \kappa_*^2 / \kappa_1^2, \end{aligned}$$

with the notations that

$$\begin{aligned} \mathbb{S}^{d-1} &:= \{x \in \mathbb{R}^d : \|x\|_2 = 1\}, \\ \overline{\mathbb{S}}^{d-1} &:= \{x \in \mathbb{R}^d : |x_1| = \dots = |x_p| = 1\}, \end{aligned}$$

and whose finiteness is guaranteed by that of $\|u^\top E_t\|_{\psi_2}$. Let us write

$$\begin{aligned} \gamma_1 &= (\kappa_* / \kappa_1) (\|\overline{A}\| / \rho_1)^K, \gamma_2 = \log(\rho_1^{-1}), \\ \gamma_3 &= p (\|\overline{A}\| / \rho_1)^K, \gamma_4 = \log(\rho_1^{-1}). \end{aligned}$$

Here we denote

$$\overline{A} := \begin{bmatrix} a_1 & a_2 & \dots & a_{d-1} & a_d \\ 1 & 0 & \dots & 0 & 0 \\ & & \dots & & \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix},$$

ρ_1 is a universal constant such that $\rho(\overline{A}) < \rho_1 < 1$ whose existence is guaranteed by the assumption that $\sum_{k=1}^d a_k < 1$, and K is some constant only depending on ρ_1 .

Assume further

$$\gamma_1 = O(\sqrt{r^*}) \text{ and } \gamma_3 = O(1).$$

Then, for any integer $n \geq 2$ and $0 \leq m \leq n - 1$, we have

$$\mathbb{E} \|\widehat{\Sigma}_m - \mathbb{E} \widehat{\Sigma}_m\| \leq C \kappa_1^2 \left\{ \sqrt{\frac{r_* \log ep}{n - m}} + \frac{r_* \log ep (\log np)^3}{n - m} \right\}$$

for some universal constant C .

Combining Lemma 4.1 and Theorem 4.1 directly yields the following large-sample property on approximating A using \widehat{A}_λ .

Theorem 4.2 Assume Model (4.1.18), the spectrum of $\Sigma_0 := \mathbb{E} \widehat{\Sigma}_0$ upper and lower bounded by universal positive constants, and assume the transition matrix $A \in \mathcal{A}(r, \gamma_{\max})$ with

$$\mathcal{A}_M(r, \gamma_{\max}) := \left\{ M \in \mathbb{R}^{d \times d} : \text{rank}(M) \leq r, \|M\| \leq \gamma_{\max} \right\}$$

for some universal positive constant $\gamma_{\max} < 1$. Suppose further that $\lambda \asymp \sqrt{d/n}$ and $\|u^\top E_t\|_{\psi_2} < \infty$. We then have

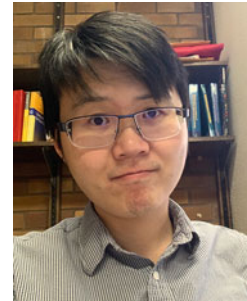
$$\|\widehat{A}_\lambda - A\|_F = O_p(\sqrt{dr \log d/n}),$$

provided that $dr \log d/n = o(1)$.

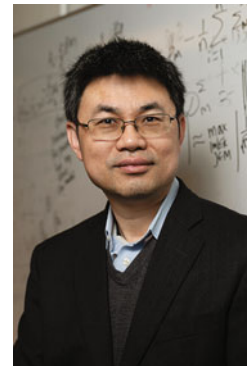
References

1. Adamczak, R.: Moment inequalities for U-statistics. *Anna. Probab.* **34**(6), 2288–2314 (2006)
2. Ahlswede, R., Winter, A.: Strong converse for identification via quantum channels. *IEEE Trans. Inf. Theory* **48**(3), 569–579 (2002)
3. Arcones, M.A., Gine, E.: Limit theorems for U-processes. *Anna. Probab.* **21**(3), 1494–1542 (1993)
4. Bai, J.: Inferential theory for factor models of large dimensions. *Econometrica* **71**(1), 135–171 (2003)
5. Bańbura, M., Giannone, D., Reichlin, L.: Large Bayesian vector auto regressions. *J. Appl. Econ.* **25**(1), 71–92 (2010)
6. Banna, M., Merlevède, F., Youssef, P.: Bernstein-type inequality for a class of dependent random matrices. *Random Matrices Theory Appl.* **5**(2), 1650006 (2016)
7. Basu, S.: Modeling and Estimation of High-dimensional Vector Autoregressions. PhD thesis, The University of Michigan, 2014
8. Bickel, P.J., Bühlmann, P.: A new mixing notion and functional central limit theorems for a sieve bootstrap in time series. *Bernoulli* **5**(3), 413–446 (1999)
9. Bradley, R.C.: Basic properties of strong mixing conditions. A survey and some open questions. *Probability Surveys* **2**(2), 107–144 (2005)
10. Bradley, R.C.: Introduction to Strong Mixing Conditions, vol. 1. Kendrick Press (2007)
11. Chan, K.-S., Tong, H.: *Chaos: A Statistical Perspective*. Springer (2001)

12. Christiano, L.J., Eichenbaum, M., Evans, C.L.: Monetary policy shocks: What have we learned and to what end? *Handbook Macroecon.* **1**(A), 65–148 (1999)
13. Dedecker, J., Prieur, C.: Coupling for τ -dependent sequences and applications. *J. Theor. Probab.* **17**(4), 861–885 (2004)
14. Dedecker, J., Prieur, C.: New dependence coefficients. examples and applications to statistics. *Probab. Theory Relat. Fields* **132**(2), 203–236 (2005)
15. Dedecker, J., Doukhan, P., Lang, G., Leon, J., Louhichi, S., Prieur, C.: *Weak Dependence: With Examples and Applications*. Springer, New York (2007)
16. Doukhan, P.: *Mixing: Properties and Examples*. Springer (1994)
17. Doukhan, P., Louhichi, S.: A new weak dependence condition and applications to moment inequalities. *Stochast. Process. Appl.* **84**(2), 313–342 (1999)
18. Doukhan, P., Neumann, M.H.: Probability and moment inequalities for sums of weakly dependent random variables, with applications. *Stochast. Process. Appl.* **117**(7), 878–903 (2007)
19. Freedman, D.A.: On tail probabilities for martingales. *Anna. Probab.* **3**(1), 100–118 (1975)
20. Giné, E., Latała, R., Zinn, J.: Exponential and moment inequalities for U -statistics. In: *High Dimensional Probability II*, vol. 47, pp. 13–38. Springer (2000)
21. Greenshtein, E., Ritov, Y.: Persistence in high-dimensional linear predictor selection and the virtue of overparametrization. *Bernoulli* **10**(6), 971–988 (2004)
22. Han, F.: An exponential inequality for U -statistics under mixing conditions. *J. Theor. Probab.* **31**(1), 556–578 (2018)
23. Han, F., Li, Y.: Moment bounds for large autocovariance matrices under dependence. *J. Theor. Probab.* **33**(3), 1445–1492 (2020)
24. Hoeffding, W.: Probability inequalities for sums of bounded random variables. *J. Am. Stat. Assoc.* **58**(301), 13–30 (1963)
25. Ibragimov, I.A., Rozanov, Y.A.: *Gaussian Random Processes*. Springer (2012)
26. Izenman, A.J.: Reduced-rank regression for the multivariate linear model. *J. Multivariate Anal.* **5**(2), 248–264 (1975)
27. Kolmogorov, A.N., Rozanov, Y.A.: On strong mixing conditions for stationary Gaussian processes. *Theory Probab. Appl.* **5**(2), 204–208 (1960)
28. Korolyuk, V.S., Borovskich, Y.V.: *Theory of U -statistics*. Springer (1994)
29. Lam, C., Yao, Q.: Factor modeling for high-dimensional time series: inference for the number of factors. *Anna. Stat.* **40**(2), 694–726 (2012)
30. Lam, C., Yao, Q., Bathia, N.: Estimation of latent factors for high-dimensional time series. *Biometrika* **98**(4), 901–918 (2011)
31. Lee, A.J.: *U -statistics: Theory and Practice*. CRC Press (1990)
32. Liebscher, E.: Towards a unified approach for proving geometric ergodicity and mixing properties of nonlinear autoregressive processes. *J. Time Series Anal.* **26**(5), 669–689 (2005)
33. Liu, W., Xiao, H., Wu, W.B.: Probability and moment inequalities under dependence. *Statist. Sinica* **23**(3), 1257–1272 (2013)
34. Mackey, L., Jordan, M.I., Chen, R.Y., Farrell, B., Tropp, J.A.: Matrix concentration inequalities via the method of exchangeable pairs. *Anna. Probab.* **42**(3), 906–945 (2014)
35. Merlevède, F., Peligrad, M., Rio, E.: Bernstein inequality and moderate deviations under strong mixing conditions. In: *High Dimensional Probability V: the Luminy Volume*, pp. 273–292. Institute of Mathematical Statistics (2009)
36. Negahban, S., Wainwright, M.J.: Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *Anna. Stat.* **39**(2), 1069–1097 (2011)
37. Oliveira, R.I.: Concentration of the adjacency matrix and of the Laplacian in random graphs with independent edges (2009). arXiv:0911.0600
38. Pan, J., Yao, Q.: Modelling multiple time series via common factors. *Biometrika* **95**(2), 365–379 (2008)
39. Rio, E.: *Asymptotic Theory of Weakly Dependent Random Processes*. Springer (2017)
40. Shen, Y., Han, F., Witten, D.: Tail behavior of dependent V -statistics and its applications. Preprint (2019). arXiv:1902.02761
41. Tropp, J.A.: User-friendly tail bounds for sums of random matrices. *Found. Comput. Math.* **12**(4), 389–434 (2012)
42. Wu, W.B.: Nonlinear system theory: Another look at dependence. *Proc. Natl. Acad. Sci.* **102**(40), 14150–14154 (2005)
43. Wu, W.B.: Asymptotic theory for stationary processes. *Stat. Interface* **4**(2), 207–226 (2011)
44. Wu, W.-B., Wu, Y.N.: Performance bounds for parameter estimates of high-dimensional linear models with correlated errors. *Electron. J. Stat.* **10**(1), 352–379 (2016)
45. Zhang, D., Wu, W.B.: Gaussian approximation for high dimensional time series. *Anna. Stat.* **45**(5), 1895–1919 (2017)



Fang Han received his Ph.D. degree in biostatistics in 2015 from the Johns Hopkins University. He is currently an associate professor in the Department of Statistics at the University of Washington, Seattle. His research interests include rank- and graph-based methods, statistical optimal transport, mixture models, nonparametric and semiparametric regressions, time series analysis, and random matrix theory.



Wei Biao Wu received the Ph.D. degree in statistics in 2001 from the University of Michigan, Ann Arbor, MI. He is currently a Professor in the Department of Statistics at the University of Chicago, Chicago, IL. His research interests include probability theory, statistics and econometrics. He is currently interested in covariance matrix estimation, regression, mean vector estimation, and multiple testing problems for dependent data.



Contents

42.1	Introduction	865
42.1.1	History.....	866
42.1.2	Machine Learning Importance.....	866
42.2	Machine Learning Categories	867
42.2.1	Types of Learning.....	867
42.2.2	Parametric Versus Nonparametric Learning.....	867
42.2.3	Deep Learning.....	868
42.3	Machine Learning Algorithms Based on Applications	868
42.3.1	Supervised Learning.....	868
42.3.2	Unsupervised Learning.....	876
42.3.3	Semisupervised Learning.....	878
42.4	Case Study	878
42.4.1	Exploratory Data Analysis (EDA).....	879
42.4.2	Modeling Selection Criteria.....	879
42.4.3	Results Analysis.....	882
42.5	Summary	883
	References	884

focus on practical algorithms of various machine learning techniques and their evolutions. An in-depth analysis and comparison based on the main concepts are presented. Different learning types are studied to investigate each technique's goals, limitations, and advantages. Moreover, a case study is presented to illustrate the concepts explained and make a practical comparison. This chapter helps researchers understand the challenges in this area, which can be turned into future research opportunities, and at the same time gain a core understanding of the most recent methodologies in machine learning.

Keywords

Machine learning · Supervised learning · Unsupervised learning · Model selection · Algorithms · Regression · Classification · Clustering

Abstract

We are living in the golden era of machine learning as it has been deployed in various applications and fields. It has become the statistical and computational principle for data processing. Despite the fact that most of the existing algorithms in machine learning have been around for decades, the area is still booming. Machine learning aims to study the theories and algorithms in statistics, computer science, optimization, and their interplay with each other. This chapter provides a comprehensive review of past and recent state-of-the-art machine learning techniques and their applications in different domains. We

42.1 Introduction

Machine learning has become one of the most exciting and popular areas in science and technology for a number of reasons. This field is rapidly expanding and tremendously fast-paced with novel algorithms and ideas introduced every day. While the concepts of machine learning have been around for decades, the progress toward developing its basis in today's technology and society is rather a new phenomenon.

In the recent decade, the growing mass of available data and invention of powerful computational processing tools has empowered machine learning and its applications significantly. Machine learning (ML) now plays an important role in today's modern life and is an indispensable part of our lives. Impacting many different businesses, we can say that machine learning and Artificial Intelligence have caused an industrial revolution by bringing many critical innovations in technology. Consider finding the fastest route to your destination using GPS navigation services such as

M. Arabzadeh Jamali (✉) · H. Pham
 Department of Industrial and Systems Engineering, Rutgers University, Piscataway, NJ, USA
 e-mail: ma.arabzade@rutgers.edu; ma1319@scarletmail.rutgers.edu; hopham@soe.rutgers.edu

Google Maps, the recommendations you receive every day in Amazon and Netflix, and Fraud alerts on your credit card, to more important applications such as Medical Diagnosis; all are applications of ML in our lives.

The main goal in machine learning is to develop computer programs and algorithms that can conduct experiments on the data and learn from it to find specific patterns. In all algorithms, the goal is to transform the raw input data to an informative output. The computer (machine) is programmed to automatically perform a model or algorithm on the input dataset in order to get the desired output. However, some of the complex learning algorithms in machine learning have completely changed our perception of data processing such as Neural Networks, boosting algorithms, and ensemble methods. This chapter presents a comprehensive literature review and future prospect of algorithms in machine learning with a multiscope perspective. In addition to in-depth analysis of algorithms, formulations, applications of machine learning, challenges, and research avenues are discussed.

42.1.1 History

Traditional statistical theories used for predictive analysis purposes before 1950s are the basis of many machine learning algorithms. However, we can say that the concept of machine learning that we know today began after the 1950s with a series of inventions, and after the 1990s, mainly the last 30 years, machine learning entered its golden era. Machine learning was first introduced by Arthur L. Samuel in the 1950s, and the term “machine learning” was defined as a “computer’s ability to learn without being explicitly programmed” [1]. He designed a computer program that could learn which moves and strategies win better in checkers and improved as it ran. In 1957, Frank Rosenblatt designed the first artificial neural network, called Perceptron, with the idea to simulate the processes of the human brain [2]. The limitations of Perceptron to recognize many classes of patterns lead to the discovery of multilayer perceptron in the 1960s. In the 1970s, the concept of Backpropagation was developed [3] and was implemented in neural networks in 1986 [4]. In 1967, the “nearest neighbor” algorithm was developed for pattern classification [5]. Bayesian methods and probabilistic inference started to appear in machine learning research in the 1980s. In the 1990s, Machine learning began focusing on more data-driven approaches instead of parametric ones. In 1995, Vapnik developed a modified classification algorithm for handling nonlinearly separable classes using Support vector machines (SVMs) [6]. Tree-based learning algorithms and Random forests were also introduced [7].

In the twenty-first century, with the invention of GPUs, which have much more processing power compared to CPUs,

along developing parallel computing frameworks and tools, scientists began developing algorithms that can analyze large amounts of data with the tools that speed up the training process. Introducing models such as Boosting algorithms, Ensemble methods, and Neural Networks which have increasing accuracy and adaptive learning pace also played an important role in ML industry in the past 20 years. ML algorithms combined with new computing technologies have boosted scalability and improve accuracy and efficiency. Nowadays, machine learning is the one of the important reasons for significant advancements in technology and has created many application-based concepts. The applications vary from predicting events and product recommendations to analyzing stocks market.

42.1.2 Machine Learning Importance

Machine learning is a dynamic domain which intersects statistics, operation research, and computer science. The mathematical theories behind the algorithms, optimization methods in hyper parameter estimation, and efficient computing frameworks are all used in ML which makes it one of the most popular fields. Despite the progress and advancements, there are still numerous challenges that machine learning faces today. High dimensional unstructured data, finding the balance point in the trade of overfitting and underfitting, and achieving the satisfying accuracy level versus expensive implementation and computational needs are among some of the challenges data scientists deal with today. Data analysts use these practical challenges and turn them into opportunities which makes machine learning a fast-evolving field. As a result, there are new methodologies and models developed by scientists every day. Machine learning is mainly related to computational statistics; however, many algorithms are driven by applications in real-world problems. How machine learning impacts many industries, and the jobs within them, has made it a competence area for researchers in all fields. Applications are varying from health care to finance and banking.

Another reason for the rapid developments of new algorithms and models in ML is the need for different algorithms based on different characteristics and the goal. Different algorithms have different efficacy on datasets and systems. For example, working with big scale unstructured data such as images or text for predicting problems, neural networks tend to outperform other algorithms. However, neural networks required high working and storing memory space. On the other hand, when it comes to structured or tabular data, decision tree-based models are better choices historically. These differences in the algorithms and also datasets and systems necessitate developing new algorithms with faster,

less memory consuming, and easier implementation models and frameworks. Here, we intend to go through the most popular algorithms in ML in both mathematical and practical aspect. In addition, various machine learning applications are reviewed to help other researchers expand their view in machine learning. In Sect. 42.2, popular machine learning categorizations by scientists are presented. Section 42.3 discusses several algorithms, techniques, and frameworks in machine learning in depth along with a number of machine learning applications. Machine learning has been used for a prediction case study to present a number of differences, advantages, and limitations of some of machine learning algorithms in Sect. 42.4. Finally, Sect. 42.5 concludes the chapter with a summary.

42.2 Machine Learning Categories

42.2.1 Types of Learning

Most common categorization of machine learning algorithms is based on their learning type. Algorithms learning types can be supervised, semisupervised, unsupervised, and reinforcement learning. In the proceeding, we go through what each learning type means and investigate the most common algorithms in their domain.

Supervised Learning

Supervised learning aims to predict the label or outcome (dependent variable) based on a set of predictors (independent variables). The algorithm generates a mapping function that takes the input data or training set and learns to map it to the label or targeted variable (e.g., true/false). This function can be used to predict the label of new or unseen data. Most important property of supervised learning is that the dataset is a collection of labeled examples $\{(x_i, y_i)\}$ where x_i 's are the predictors, independent variables, or features vector and y_i is the label, response, or targeted variable. The goal of supervised learning algorithms is to take labeled training data and a set of features vector x as the input and output a model that predicts the label of the testing set y_i .

Unsupervised Learning

Contrary to supervised learning, in unsupervised learning the input datasets are not labeled. This learning type can be used when we only have input data x with no output variables or labels corresponding to the x_i 's. Giving no labels to the learning algorithm, a function is designed to find the patterns in the datasets. Considering x as the feature vector, the main goal in unsupervised learning is to produce a model that takes this vector x as input and transforms it into an informative value that can be used to find the structure of new unseen datasets.

Semisupervised Learning

Semisupervised learning is a mixture of supervised and unsupervised learning meaning the dataset contains both labeled and unlabeled data y 's. The goal in semisupervised learning is the same as the goal of the supervised learning algorithms but with the difference that an incomplete training dataset with missing target outputs is available. Semisupervised learning is often used in situations where the cost associated with the labeling process is expensive, whereas unlabeled data is relatively inexpensive, and semisupervised learning can help the learning process to result in a more accurate model.

Reinforcement Learning

Reinforcement learning is different than the three previous learning types in the way that the machine learns from the predefined actions and rewards associated with them. Here the machine must take actions in a dynamic environment in order to gain as many rewards as possible using training data that is given as feedbacks to the machine's actions [8]. Reinforcement learning is primarily based on Markov decision process, and the algorithm learns to choose actions in different situations to maximize the reward or feedback signal. Furthermore, the environment is formulated as a Markov decision process, situations are defined as different states in the state space, and the machine can execute actions in every state. The algorithm is trained to learn and find the most rewarding actions by trial and error. Each action brings different rewards and moves the machine to another state of the environment. The goal of a reinforcement learning algorithm is to learn to map actions to the proceeding states and their corresponding reward. It takes the feature vector of a state as input and outputs an optimal action to execute in that state by maximizing the expected average reward. Reinforcement learning is useful in sequential decision-making problems with long-term objectives and therefore has applications in domains such as robotics, resource management, and logistics. Another interesting topic that has been recently introduced is deep reinforcement learning in which deep neural network is implemented to avoid the need of explicitly designing the state space [9, 10]. Most famous algorithms used in Reinforcement learning are Q-learning, Deep Q networks, Deep Deterministic Policy Gradient (DDPG), and State-Action-Reward-State-Action (SARSA) [11–15].

42.2.2 Parametric Versus Nonparametric Learning

Learning algorithms can be parametric or nonparametric. For instance, most supervised learning algorithms are parametric or model based. Parametric algorithms use the training data to create a model that has parameters learned from the training data. On the other hand, nonparametric or instance-based

learning algorithms use the whole dataset to fit a model with no specified parameters. One example of these algorithms are instance-based classifiers, also called lazy learners, that store all of the training samples and do not build a classifier until a new, unlabeled sample needs to be classified. Consider Regression problems, the most popular nonparametric algorithms include K -Nearest Neighbors (KNN), Gaussian Process, and Random Forests. Parametric algorithms used for such problems include Ordinary Least Squares, MARS, and ridge regression.

42.2.3 Deep Learning

In the conventional machine learning, the parameters of the model are directly learnt from the dataset which is also known as shallow learning. Deep learning, on the other hand, is based on multilayer artificial neural networks and have high level of complexity [16]. In deep learning, most model parameters are learned not directly from the features of the training instances, but rather from the outputs of the inner or hidden layers of an artificial neural networks. In this case, we can consider shallow learning as the output of a single layer feed forward network. These concepts will be discussed more in the artificial neural networks chapter.

Deep learning algorithms such as deep belief networks (DBN), recurrent neural networks (RNN) [17], convolutional neural networks (CNN) [18, 19] and Auto-Encoders [20, 21] have shown high performances in many applications such as computer vision and image processing, speech recognition, natural language processing, and audio recognition [15, 22]. Generative adversarial network (GAN) is one of the newest concepts in deep learning introduced in recent years which outperforms many techniques in areas such as computer vision [23].

42.3 Machine Learning Algorithms Based on Applications

Another categorization of machine learning task arises when one considers the desired output of a machine-learned system. Classification and prediction are two forms of data analysis that can be used to extract models describing important data classes or to predict future data trends. In what follows, we will go through most popular ML techniques and categorize them based on their applications, linearity or nonlinearity, accuracy, and preference over each other in different situations. One of the challenges data analysts face at the beginning is to use a linear model or nonlinear one in their analysis. Despite the fact that nowadays there are many nonlinear models to use which usually give you high accuracy, many people still prefer linear models because of

their computational simplicity and interpretability. Moreover, it has been shown that linear models tend to perform better with small sample sizes.

Bias – Variance One of the important concepts in ML is bias-variance trade-off in estimated models. The Mean Square Error (MSE) can be decomposed into three terms, two of which are in the hands of analyst.

$$\begin{aligned} \text{MSE}(x_0) &= E[f(x_0) - \hat{y}_0]^2 = E[\hat{y}_0 - E(\hat{y}_0)]^2 \\ &+ E[E(\hat{y}_0) - f(x_0)]^2 = \text{Var}(\hat{y}_0) + \text{Bias}^2(\hat{y}_0) + \varepsilon \end{aligned} \quad (42.1)$$

Flexible algorithms such as K -Nearest Neighbors will give a high-variance but low-bias model whereas nonflexible algorithms like least square regression will give a low-variance, but high bias model compared to the flexible ones.

42.3.1 Supervised Learning

Classification Algorithms

Classification is a problem of automatically predicting the label or class of a data point. Spam detection in emails is a famous example of classification. In machine learning, the classification problem can be solved by different classification learning algorithms that they all take a collection of labeled examples (training dataset) as inputs and create a model that can take an unlabeled example as input and either directly output a label or class or they give a value that can be used by the analysts to deduce the output label such as the probability of being in each class. There are two main approaches for classification among statisticians; the generative approach and the discriminative approach. Given an independent variable X and a target variable Y , Ng and Jordan (2002) distinguish the two categories as follows: A generative classification model is a statistical model based on the joint probability distribution of $X \times Y$, $P(x, y)$. A discriminative classification model is based on the conditional probability of the Y given an observation x , $P(Y|X = x)$ [24]. Examples of generative classifiers are Naive Bayes classifier and Linear Discriminant Analysis (LDA). And nonmodel classifier or loss discriminative is support vector machine. The most famous and used algorithms in classification problems are discussed in the following.

K -Nearest Neighbors (KNN)

A nonparametric, instance-based learning method finds the k closest observations in the training set (x_i) in input space to the new observation (x) and averages their responses (y_i) in

order to find predicted response \hat{Y} :

$$\hat{Y}(x) = \frac{1}{K} \sum_{x_i \in N_k(x)} y_i \tag{42.2}$$

where $N_k(x)$ is the k closest neighbors of x in the training sample and Euclidean distance is used mostly as the measure of closeness. In regression problems, \hat{Y} is the average of responses (y_i 's), and if it is a classification problem, \hat{Y} is the majority vote. KNN will provide a nonlinear decision boundary (nonlinear regression fit) for observations in different classes. Choosing k to be small (e.g., $k = 1$) will cause the model to have a high variance but low bias. On the other hand, if k is large (e.g., $k = 10$), a smoother decision boundary will be obtained with low variance but rather higher bias than the first case. Moreover, as dimension increases, MSE tends to increase as dimension increases. The increase in the bias term, and not the variance, is the reason for that since when dimension increases the average distance to nearest neighbors increases and causes the estimate to be bias.

Support Vector Machines (SVM)

This method classifies observations by constructing an optimal separating hyperplane between two classes. Two cases can happen: Either two classes are linearly separable or nonlinear separable, where the classes overlap. In both cases, SVM and its extension to nonlinear decision boundary can be used.

Linear Case Considering our training dataset consists of $\{x_i, y_i\}$ where y_i is either $+1$ or -1 , we define a hyperplane by

$$\{x : f(x) = x^T \beta + \beta_0\} \tag{42.3}$$

And then a classification rule is set by

$$G(x) = \text{sign} [x^T \beta + \beta_0] \tag{42.4}$$

Here we want to find the hyperplane which creates the biggest margin between training points of class $+1$ and -1 . Here we want to maximize the marginal which can be rephrased as

$$M = \frac{1}{\|\beta\|} \tag{42.5}$$

In this way, the optimization problem becomes

$$\min_{\beta, \beta_0, \|\beta\|=1} \|\beta\| \tag{42.6}$$

$$\text{subject to } y_i (x_i^T \beta + \beta_0) \geq 1, \quad i = 1, \dots, N \tag{42.7}$$

Nonlinear Case If the classes overlap in feature space so that we cannot separate them using linear hyperplane, we can use a modified version of the linear case in such way that we not only still maximize M but also allow some instances to be on the wrong side of the margin. In this way, we need to define slack variables ξ and modify the optimization problem.

$$\min_{\beta, \beta_0} \|\beta\| \tag{42.8}$$

$$\text{subject to } y_i (x_i^T \beta + \beta_0) \geq 1 - \xi_i, \quad i = 1, \dots, N \tag{42.9}$$

$$\xi_i \geq 0, \quad \sum \xi_i \leq \text{constant} \tag{42.10}$$

Probabilistic Models

LDA (Linear Discriminant Analysis)

LDA is based on Bayes' Theorem for classifications in such a way that the Bayes' classifier chooses the class with the highest posterior probability for minimum error. If we consider $f_k(x)$ as the class conditional density of X in class k and π_k as the prior probability of class k , the theorem gives

$$\begin{aligned} \widehat{\text{Pr}}(G = k | X = x) &= \frac{\widehat{\text{Pr}}(X = x | G = k) \widehat{\text{Pr}}(G = k)}{\sum_{l=1}^K \widehat{\text{Pr}}(X = x | G = l) \text{Pr}(G = l)} \\ &= \frac{f_k(x) \pi_k}{\sum_{l=1}^K f_l(x) \pi_l} \end{aligned} \tag{42.11}$$

Considering the class densities as multivariate Gaussian distribution with common covariance matrix Σ , μ_k the mean of the input variables for class k , and p as the number of variables

$$\widehat{\text{Pr}}(X = x | G = k) = \hat{f}_k(x) \tag{42.12}$$

$$\widehat{\text{Pr}}(G = k) = \hat{\pi}_k \tag{42.13}$$

$$f_k(x) = \frac{1}{(2\pi)^{\frac{p}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1} (x-\mu_k)} \tag{42.14}$$

We can get the linear discriminant functions δ_k

$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log \pi_k \quad (42.15)$$

which are equivalent to descriptions of the decision rule so that

$$G(x) = \operatorname{argmax}_k \delta_k(x) \quad (42.16)$$

π_k is estimated as the fraction of training samples of class k .

Considering we have only two classes, the LDA rule classifies a new observation x to class 2 if

$$x^T \Sigma^{-1} (\hat{\mu}_2 - \hat{\mu}_1) > \frac{1}{2} (\hat{\mu}_2 + \hat{\mu}_1)^T \Sigma^{-1} (\hat{\mu}_2 - \hat{\mu}_1) + \log \left(\frac{N_2}{N_1} \right) \quad (42.17)$$

QDA (Quadratic Discriminant Analysis)

LDA was a special case when the density functions have common covariance matrix Σ . If Σ_k are not assumed to be equal, we get quadratic discriminate function

$$\delta_k(x) = -\frac{1}{2} \log |\Sigma_k| - \frac{1}{2} (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) + \log \pi_k \quad (42.18)$$

Logistic Regression

Logistic regression is a linear method, for that classifies observations based on conditional probabilities and gives the posterior probabilities of classes using linear function in x . It has the form of $(K - 1)$ log-odds or logit transformations.

$$\log \frac{\Pr(G = i | X = x)}{\Pr(G = K | X = x)} = \beta_{i0} + \beta_i^T x, \quad i = 1, \dots, K - 1 \quad (42.19)$$

which is equivalent to

$$\begin{aligned} \Pr(G = i | X = x) \\ = \frac{\exp(\beta_{i0} + \beta_i^T x)}{1 + \sum_{l=1}^{K-1} \exp(\beta_{l0} + \beta_l^T x)}, \quad i = 1, \dots, K - 1 \end{aligned} \quad (42.20)$$

$$\Pr(G = K | X = x) = \frac{1}{1 + \sum_{l=1}^{K-1} \exp(\beta_{l0} + \beta_l^T x)} \quad (42.21)$$

Here the last class, class K , has been used as the denominator in the odds-ratio, but in general it could be arbitrary.

Additionally, we should keep in mind that the sum of the probabilities should be equal to 1.

Naïve Bayes Classifier

Naïve Bayes algorithm estimates the class conditional probability density function (pdf) by assuming that the features X_i 's are conditionally independent given the class label. If we consider $f_k(x)$ as the class conditional density of X in class k , we can find it by:

$$f_k(X) = \prod_{i=1}^p f_{ki}(X_i) \quad (42.22)$$

The assumption of independency among features is generally not true, but it simplifies the estimation specially when p or the number of dimensions is high.

Tree-Structured Learning

Classification and Regression Tree (CART)

Decision tree method consists of first dividing the feature (predictors) space into a number of regions or rectangles by a set of recursive binary splitting rules and then fitting a simple regression or classification model in each one. In order to make predictions for new observations in classification and regression tree, known as CART, we usually take the mean (regression setting) or majority votes (classification setting) of trainings observations in the region it belongs to. Your prediction values are at terminal nodes of leaves. ID3 (Iterative Dichotomize 3) was the first Decision Tree implementations developed by Ross Quinlan 1986. C4.5 is the successor to ID3 which did not have the restriction that features must be categorical, and C5.0 is the most recent algorithm developed by Quinlan [25, 26]. Decision trees are simple and easy to interpret but not comparable to other methods in terms of accuracy. Many advanced and powerful algorithms are based on combining many decision trees to get the better prediction accuracy. Considering a regression problem with R_1, \dots, R_M regions, the predictions y_i 's in each region will be a constant c_m which is the average of response values in that region.

$$f(x) = \sum_{m=1}^M c_m I(x \in R_m) \quad (42.23)$$

$$\hat{c}_m = \frac{1}{N_m} \sum_{x_i \in R_m} y_i \quad (42.24)$$

where N_m is instances in training set that belong to region R_m

$$N_m = \sum_{i=1}^N I(x_i \in R_m) \tag{42.25}$$

The challenge in decision tree is how to grow the tree. In other words, what is the best partition regions of input space means deciding the splitting variable and split point at each step of the tree. Considering variable j and split point s if we define regions as

$$R_1(j, s) = \{X|X_j \leq s\} \quad \text{and} \quad R_2(j, s) = \{X|X_j > s\}, \tag{42.26}$$

we find the best split by the following optimization problem

$$\min_{j,s} \left[\min_{c_1} \sum_{x_i \in R_1(j,s)} (y_i - \hat{c}_1)^2 + \min_{c_2} \sum_{x_i \in R_2(j,s)} (y_i - \hat{c}_2)^2 \right] \tag{42.27}$$

In decision trees, the tree size is a tuning parameter that needs to be optimized. In order to avoid overfitting, the idea is to grow a really deep tree, T_0 , until some minimum node size is reached, and then prune it by collapsing some internal nodes so that it meets a certain cost complexity.

If we define a subtree $T \subset T_0$, we will have the within-node squared error as

$$Q_m(T) = \frac{1}{N_m} \sum_{x_i \in R_m} (y_i - \hat{c}_m)^2 \tag{42.28}$$

And define cost complexity as

$$C_\alpha(T) = \sum_{m=1}^T N_m Q_m(T) + \alpha |T| \tag{42.29}$$

Where $|T|$ is the tree size or number of terminal nodes in the tree T , and we try to select the subtree with minimum C_α .

In classification setting, the squared error impurity measure cannot be used. If we consider the proportion of class k observations in region m as

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k), \tag{42.30}$$

we classify the observations in node m to the majority class in node m

$$k(m) = \operatorname{argmax}_k \hat{p}_{mk} \tag{42.31}$$

The three impurity measures for classification are
Misclassification error:

$$\frac{1}{N_m} \sum_{x_i \in R_m} I(y_i \neq k_m) = 1 - \hat{p}_{mk(m)} \tag{42.32}$$

Gini index:

$$\sum_{k=1}^K \hat{p}_{mk} (1 - \hat{p}_{mk}) \tag{42.33}$$

Cross-entropy or deviance:

$$-\sum_{k=1}^K \hat{p}_{mk} \log \hat{p}_{mk} \tag{42.34}$$

There is a special case when $k = 2$; Gini index will be the sum of all variances which is the same as the impurity measure in regression setting. Moreover, Gini index and cross-entropy are sensitive to different splits unlike misclassification error. In this regard, misclassification error is not really used for growing the tree among all measures. Gini index is also computationally preferred to cross-entropy. Usually, Gini index is used to grow the trees since we want each group in each split node to be as pure as possible, and then misclassification error and cross-entropy are referred for performance evaluation. There are some types of functions and datasets that CART does have trouble approximating. Additive functions, functions with categorical variables, and datasets with many missing observations are among them.

Bagging

Bagging averages predictions of bootstrap samples in order to reduce the variance in expected error. For each bootstrap sample Z^b , $b = 1, \dots, B$, we can get the prediction $\hat{f}^b(x)$, and the bagged estimate will be:

$$\hat{f}_{\text{bag}}(x) = \frac{1}{B} \sum_{b=1}^B \hat{f}^b(x) \tag{42.35}$$

In Bagging, bootstrap samples of original training set are used to make sure sample predictions are independent of each other. Bagging is especially useful in prediction methods with high variance and unstable results like decision trees as it lowers variance and improves prediction. However, if the predictions are correlated, the variance will rise. That is one of the reasons random forest was invented.

Random Forest

As previously mentioned, because of the correlation between trees in bagging method, the alternative, random forests, came up to use uncorrelated trees and averages them. The major modification is that rather than choosing the best j variable among all variables x_1, \dots, x_p to grow the tree, we

choose a subset M of size m randomly from $\{x_1, \dots, x_p\}$ at each step of the tree. This way, the trees become much more uncorrelated. In fact, the higher the number of variables in the subset m , the higher chance of correlation between variables. Therefore, a smaller subset of m will more likely give uncorrelated trees. In random forest, we draw a bootstrap sample Z^b , $b = 1, \dots, B$, from the training and grow a random forest T_b to the dataset by selecting m variables randomly at each node.

$$\hat{f}_{\text{rf}}(x) = \frac{1}{B} \sum_{b=1}^B T_b(x) \quad (42.36)$$

The default values for m , for regression problems, are usually $\lfloor \frac{p}{3} \rfloor$ and $\lfloor \sqrt{p} \rfloor$ for classification problems. One important aspect in random forest and bagging is the use of Out-of-Bag samples. Since in bagging and random forest the trees are fit to bootstrapped samples of observations, one can show that approximately 30% of your data will not be selected in the samples structuring the trees. This way, we get the out-of-bag (OOB) error rate by predicting the response for the observations by using only the trees that were not fit using those observations and were OOB. The OOB error rate is equivalent to the test error rate in cross-validation approach. Mean Decrease in Impurity (MDI) and Mean Decrease in Accuracy (MDA) are used for variable importance measure and model selection.

Boosting Methods

The main idea in boosting is to combine many so-called “weak learners” or base learners (decision stumps) to produce a powerful model. “weak learner” is defined as one whose error rate is slightly better than random guessing. In this respect, it is similar to Bagging and other committee-based approaches. In this study, most known boosting algorithms will be reviewed.

AdaBoost

Considering a classification problem $Y \in \{1, -1\}$, boosting applies weak classification algorithm and iteratively changes weights of data points in training set so that at each round more attention is paid to the misclassified ones. This way, a sequence of weak classifiers $G_m(x)$, $m = 1, 2, \dots, M$ is produced, and the prediction from all of them combined with a weighted majority vote (contribution of each $G_m(x)$) gives the final prediction $G(x)$.

$$G(x) = \text{sign} \left[\sum_{m=1}^M \alpha_m G_m(x) \right] \quad (42.37)$$

Here, α_m 's are the weights in order to increase the influence of more accurate classifiers in sequence. We initialize

the algorithms with all data points (x_i, y_i) , $i = 1, \dots, N$ having the same weight $w_i = \frac{1}{N}$, and then from $m = 2$ to M , we proceed by assigning more weights α_m , to the data points that were misclassified in the previous step according to err_m , and train the classifier $G_m(x)$.

$$\text{err}_m = \frac{\sum_{i=1}^N w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^N w_i} \quad (42.38)$$

And

$$\alpha_m = \log \left(\frac{1 - \text{err}_m}{\text{err}_m} \right) \quad (42.39)$$

And weights for next iteration classifier $G_{m+1}(x)$ are computed by

$$w_i = w_i \cdot \exp[\alpha_m I(y_i \neq G_m(x_i))], \quad i = 1, 2, \dots, N \quad (42.40)$$

Generally speaking, AdaBoost fits an additive model using an exponential loss function, and the population minimizer of this exponential loss function is the log-odds of the class probabilities. Decision trees are found to be the ideal weak learners or base learners for boosting. However, in some situation where the data is noisy(outliers) or there is a misspecification of class labels in the training set, the performance of AdaBoost (generally exponential loss function) degrades and is far less robust.

Gradient Boosting

Gradient boosting algorithm is a forward stepwise procedure that solves an optimization problem to find the next tree given the current model $f_{m-1}(x)$, and it incorporates gradient descent into it.

At each iteration, a weak learner $h(x)$ (typically a regression tree) whose predictions are as close as possible to the current residuals (negative gradient values) of the loss function is added, as if in each iteration f_{m+1} tries to correct the errors of f_m .

$$f_{m+1} = f_m(x) + h(x) = y \quad (42.41)$$

$$h(x) = f_m(x) - y \quad (42.42)$$

This is of course an approximation algorithm in order to find a good numerical result. This loss function could be squared error, absolute error, or Huber for regression and multinomial deviance for classification. In this algorithm, we first initialize the model by the optimal constant model which is a single terminal node tree.

$$f_0(x) = \operatorname{argmin}_\gamma \sum_{i=1}^N L(y_i, \gamma) \quad (42.43)$$

$$L(q) = \min \sum_{i=1}^N L(y_i, f_{m-1}(x_i) + h_m(x_i)) + \sum_{m=1}^M \Omega(h_m) \quad (42.50)$$

Where N is the number of observations in the training dataset $\{(x_i, y_i)\}$, and L is the chosen loss function. And for iteration $m = 1$ to M , we calculate the pseudoresiduals by:

$$r_{im} = - \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right]_{f=f_{m-1}} \quad (42.44)$$

A base learner $h_m(x)$ is fitted to pseudoresiduals using the training set afterward, followed by computing the multiplier γ_m (weight or proportion of $h_m(x)$ to be added to $f_{m-1}(x)$) by solving the following optimization problem:

$$\gamma_m = \operatorname{argmin}_\gamma \sum_{i=1}^N L(y_i, f_{m-1}(x_i) + \gamma h_m(x_i)) \quad (42.45)$$

And update the model:

$$f_m(x) = f_{m-1}(x) + \gamma_m h_m(x) \quad (42.46)$$

The final output is the prediction estimate

$$\hat{f}(x) = f_M(x) \quad (42.47)$$

Gradient Boosting Machine (Gradient Tree Boosting)

We can use this method using regression tree as our base learners (for both classification and regression problem) also known as MART (Multiple Additive Regression Trees) or GBM (gradient boosting machine); r_{jm} gives us the regions R_{jm} , and $h(x)$ becomes:

$$h_m(x) = I(x \in R_{jm}) \quad (42.48)$$

Consequently,

$$f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \in R_{jm}) \quad (42.49)$$

There are two tuning parameters, number of iteration M and size of each constituent tree $J_m = 1, \dots, M$.

XGBoost

XGBoost method is an implementation of gradient boosting algorithm with additional regularized loss (objective) function in order to counter overfitting.

where the added term Ω penalizes the complexity of the model and is defined as $\Omega(h) = \gamma T + \frac{1}{2} \lambda \|w\|^2$, T is the number of leaves in the tree, and q is the tree structure where each leaf in the tree has a score w_j , and the final prediction is found by summing up the score in the corresponding leaves.

After expanding the loss function similar to Taylor series, we can find the optimal weight w_j^* of leaf j and the loss function:

$$w_j^* = - \frac{\sum_{i \in I_j} g_i}{\sum_{i \in R_m} h_i + \lambda} \quad (42.51)$$

$$L(q) = - \frac{1}{2} \sum_{j=1}^T \frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \quad (42.52)$$

where g_i and h_i are the first- and second-order gradient statistics on the loss function L and I_j is the instance set in leaf j . This last equation is used as a scoring function in order to find the tree structure q by measuring the gain in different splitting nodes. An approximate algorithm that starts with a leaf and adds branches based on the gain is implemented in XGBoost. The Gain when a leaf is split into two leaves, where instance sets of the Left node I_L and Right node I_R and $I = I_R \cup I_L$, is calculated as (loss before splitting – loss after splitting), and $L(q)$ becomes:

$$L_{\text{split}} = \frac{1}{2} \left[\frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \quad (42.53)$$

If the gain $< \gamma$, that branch will not be added to the model and can be removed from the tree. Furthermore, XGBoost uses two additional techniques in order to avoid overfitting, column (feature) subsampling that speeds up parallel computation, and shrinkage introduced by Friedman. For the split-finding algorithms, XGBoost uses an approximate algorithm instead of the common exact greedy algorithm; that is because the exact algorithm is computationally expensive and impossible when the data does not fit into memory. In the approximate algorithm, candidate splitting points are chosen based on the percentiles of feature distribution, and then based on the aggregated statistics the best solution among the candidates is found. In order to do the approximate algorithm more efficiently, they use column block for parallel learning which means collecting statistics for each column (feature)

will be parallelized, giving us parallel algorithm for split finding. Additionally, the column block technique also enables column subsampling which was mentioned previously.

As previously mentioned, it has been seen that decision trees do poorly with sparse data. In this regard, XGBoosting suggests an improved sparsity-aware split-finding technique in order to make the algorithm aware of the sparsity pattern in the data. They propose a default direction in each splitting node such that if a value is missing, the instance goes to the default direction. The optimal directions are learnt from the data and set as the default direction in each branch, and the big improvement then would be that the algorithm only visits the nonmissing entries. This way, the sparsity aware algorithm runs 50 times faster than the naïve version.

XGBoost might be preferred to some competitive algorithms such as Neural network because of fewer number of parameters to tune, and also it is easy to implement. Light GBM and Catboost are also new boosting algorithms that came out in 2017 as a modification of XGboost [27, 28].

Artificial Neural Networks (ANN)

Neural network is different than the previous methodologies as it constructs a framework that investigates underlying behavioral trends of the system. Neural network recognizes the hidden relationships between input variables and the relationship between input and output variables that helps making more accurate predictions [29]. Neural networks are capable of learning complicated nonlinear relationships and capture complex data patterns and hidden features. These properties make them preferable for classification and prediction problems that involve the detection of complicated nonlinear patterns specially in high-dimensional datasets.

Perceptron learning is the foundation for the neural network models introduced by Rosenblatt in 1958 [30]. A Perceptron in neural network is an artificial neuron that performs certain functions to the input data in order to generate features. In neural network, we can have single-layer Perceptron which can only learn linearly separable patterns or multilayer Perceptron that have more than two layers and perform better on more complex problems.

Neural networks use the Kolmogorov-Arnold representation theorem that any function $f(x)$ can be exactly represented by a (possibly infinite) sequence of addition, multiplication, and composition with functions that are universal and do not depend on $f(x)$.

$$f(x) = \sum_{m=1}^M g_m(w_m^T x) \quad (42.54)$$

Here the g_m or g function is called the activation function which does not depend on M and is typically taken to be sigmoid function. However, hyperbolic tangent and rectified

linear unit (ReLU) are among the most popular ones now in industry. W_m is called the inner-layer or hidden layer weight vector, and M is the width or number of hidden units ($w_m^T x$). In order to find the Risk function or the deviance, we use gradient descent; the most popular learning technique in neural network is back-propagation.

$$\hat{R} = \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K (y_{ik} - f_k(x_i))^2 \quad (42.55)$$

where $f(x)$ has the form of

$$f(x) = \sum_{m=1}^M \beta_m \sigma(\alpha_{0m} + \alpha_m^T x) \quad (42.56)$$

Neural network uses a set of chain rules in order to find the minimum of R function. The parameters in this equation that need to be optimized can be found with backpropagation equations

$$\frac{\partial R_i}{\partial \beta_{km}} = -2 (y_i - f_k(x_i)) g_k^T(T) Z_{mi} \quad (42.57)$$

$$\frac{\partial R_i}{\partial \alpha_{ml}} = -2 (y_i - f_k(x_i)) g_k'(T) \beta_{km} \sigma'(\alpha_{0m} + \alpha_m^T x_i) x_{il} \quad (42.58)$$

where

$$Z_{mi} = \sigma(\alpha_{0m} + \alpha_m^T x_i) \quad (42.59)$$

One of the disadvantages of neural networks is that it is very subjective, where the model and results vary significantly on the data analyst network configuring and architecture, initialization, and parameter tuning. The problem may rise as there are many parameters to be optimized in NN, and overfitting might happen. In this regard, usually scientists initialize the parameters with standard normal distribution and add a penalization term to the risk function.

There are multiple neural network types, some of which are most popular among researchers. Feedforward Neural Network is the simplest and easiest type of neural network in which the information can only move from the input layer through the hidden layers and the output layer. On the other hand, in Recurrent Neural Network (RNN) the output of a layer is fed back to the input in order to improve the final predictions. Convolutional Neural Networks (CNN) are similar to feed forward neural networks but contain one or more convolutional layers in which a convolutional operation is performed to the input before passing to the next layer. CNNs show highly competitive results in image and video

recognition and natural language processing. Auto-Encoders are another popular type of neural networks that are used in unsupervised learning, specifically dimension reduction and denoising. Further discussion about neural network and deep learning is out of scope of this chapter but can be found in references [31, 32].

Regression Algorithms

Regression is a problem of predicting a real-valued label (often called a target) given an unlabeled example. The regression problem is solved by a regression learning algorithm that takes a collection of labeled examples as inputs and produces a model that can take an unlabeled example as input and output a target.

Linear Regression Methods

Least Squares Linear Regression

Least Squares Linear regression is a popular regression algorithm that learns a model in which target value is a linear combination of features of the input observation. Least Squares Linear regression is widely used because of computational simplicity and good estimation for small sample sizes. Suppose we have N observations $\{(x_i, y_i)\}$, and we want to fit a linear regression function $f(x_i)$ to our feature vector in order to predict value of y_i :

$$Y = f(x) + \varepsilon \tag{42.60}$$

where ε is observed as independent and identically distributed random variable with mean zero and constant variance σ^2 . In other words, we want to estimate the function:

$$f(x) = E\{Y|X = x\} = \beta_0 + \sum_{j=1}^p \beta_j X_j \tag{42.61}$$

where $X = (X_1, \dots, X_p)^T$ and p is the number of predictors x . Least square method is used to estimate β 's by minimizing the Residual Sum of Square Errors:

$$\begin{aligned} \text{RSS}(\beta) &= \frac{1}{N} \sum_{i=1}^N (y_i - \beta^T x_i)^2 \\ &= \frac{1}{N} \|Y - X\beta\|^2 \text{ (in the matrix format)} \end{aligned} \tag{42.62}$$

where $\beta = (\beta_0, \beta_1, \dots, \beta_p) \in \mathbb{R}^{p+1}$. If N is greater than p , this equation is solvable, but if N is smaller than p , we cannot find a unique solution and we should do regularization. Least square method solution for β 's is:

$$\hat{\beta}_{LS} = (X^T X)^{-1} X^T Y \tag{42.63}$$

Based on the Gauss–Markov theorem, least square is the best linear unbiased estimator among all estimations. Now suppose we want a prediction at a new input X_0 :

$$Y_0 = f(X_0) + \varepsilon_0 \tag{42.64}$$

$$\tilde{f}(X_0) = X_0^T \tilde{\beta}_0 \tag{42.65}$$

We can find our prediction accuracy using expected prediction error:

$$E\left[\left(Y_0 - \hat{f}(X_0)\right)^2\right] = \text{Var}(\varepsilon_0) + \text{MSE}(\tilde{f}(X_0)) \tag{42.66}$$

Shrinkage Methods via Regularization

Overfitting is a common issue in machine learning problems and among different approaches to overcome this issue. Regularization is the most popular approach among researchers in order to resolve and avoid overfitting. Regularized Regression methods retain a subset of the predictors and regularize or shrink the remaining predictor's coefficient estimates toward zero or excluding them from the model. This can be done by adding a penalty term to the lost (cost) function.

Ridge Regression

Ridge regression coefficients minimize a penalized residual sum of squares, and λ is the parameter for controlling the shrinkage. If λ is zero, then Ridge regression is the same as ordinary sum of squares regression, and if we choose a large λ , there will be more shrinkage. Ridge regression enforces the β coefficients toward zero but does not set them to zero, which means it does not omit insignificant features but rather minimizes their impact on the model.

$$\text{RSS}(\beta) = \frac{1}{N} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \cdot \text{penalty}(\beta) \tag{42.67}$$

$$\text{penalty}(\beta) = \sum_{j=1}^p \beta_j^2 \tag{42.68}$$

$$\text{writing } B \text{ Ridge in terms of } B \text{ Least square} = \frac{1}{\lambda + 1} \hat{\beta}_{LS} \tag{42.69}$$

LASSO

Lasso is the same as ridge regression but uses L1 regularization instead of L2. However, Lasso sets coefficients of irrelevant features to zero instead of minimizing them. This is one of Lasso's advantages because this way fewer features are included in the model, meaning a subset selection and coefficient estimation at the same time.

$$\text{RSS}(\beta) = \frac{1}{N} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \cdot \text{penalty}(\beta) \quad (42.70)$$

$$\text{penalty}(\beta) = \sum_{j=1}^p |\beta_j| \quad (42.71)$$

writing BLASSO in terms of B Least square

$$= \left(\text{sign} \hat{\beta}_{ls} \right) \left(\left| \hat{\beta}_{ls} \right| - \frac{\lambda}{2} \right) \quad (42.72)$$

Elastic Net

Elastic net is the combination of Ridge and LASSO

$$\text{RSS}(\beta) = \frac{1}{N} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \cdot \text{penalty}(\beta) \quad (42.73)$$

$$\text{penalty}(\beta) = \lambda \left(\alpha \sum_{j=1}^p |\beta_j|^2 + (1 - \alpha) \sum_{j=1}^p |\beta_j| \right) \quad (42.74)$$

where α is the mixing parameter of ridge and lasso. When $\alpha = 1$, it becomes fully ridge regression, and when $\alpha = 0$, it becomes Lasso.

Support Vector Regression (SVR)

Support Vector Machine for Regression (SVR) is the same as SVM for classification setting but with little modification. The objective in SVM is to find a hyperplane that separates the two classes with the minimum error in a way that the distance between the training data points from either of the two classes is maximized. However, the goal in SVR is to find a hyperplane (function $f(x)$) with points on both sides but deviates from y_n by errors that do not exceed the threshold ε for each training point x . Considering our training, dataset consists of $\{x_i, y_i\}$ where y_i is continued:

$$\{x : f(x) = x^T \beta + \beta_0\} \quad (42.75)$$

The optimization problem becomes

$$\min_{\beta, \beta_0, \|\beta\|=1} \|\beta\| \quad (42.76)$$

$$\text{subject to} \quad |y_i - (x_i^T \beta + \beta_0)| \leq \varepsilon, \quad i = 1, \dots, N \quad (42.77)$$

In order to overcome infeasible constraints of the optimization problem, formulation is modified with slack variables ξ_i, ξ_i^* :

$$\min_{\beta, \beta_0} \|\beta\| + C \sum_{i=1}^N \xi_i + \xi_i^* \quad (42.78)$$

$$\text{subject to} \quad y_i - (x_i^T \beta + \beta_0) \leq \varepsilon + \xi_i \quad (42.79)$$

$$(x_i^T \beta + \beta_0) - y_i \leq \varepsilon + \xi_i^*, \quad i = 1, \dots, N \quad (42.80)$$

$$\xi_i, \xi_i^* \geq 0 \quad (42.81)$$

42.3.2 Unsupervised Learning

In unsupervised learning, we are interested in estimating the probability density $p(x)$ based on a set of observations (x_1, \dots, x_N) without considering its relationship to other variables and having a “label” or associated response variable Y . In this regard, we cannot use previous loss functions like squared error to evaluate the methods. Unsupervised learning can be categorized into clustering algorithms, dimension reduction, and density estimation. There are many density estimation methodologies such as Gaussian mixture, kernel density, and deep Boltzmann machine, but it is a broad topic by itself and cannot be covered in this chapter. Some researchers categorize kernel methods as the fourth group in unsupervised learning.

Clustering Analysis

Clustering is related to grouping or segmenting the dataset into subsets or clusters in order to present $p(x)$ by simpler densities in a way that observations within each cluster are more closely related to one another. One of the main applications in clustering is to perform market segmentation. Similarity/dissimilarity is the metric that shows how close or far the data points are from clusters and each other in clustering. Each clustering method attempts to group the dataset points based on the dissimilarity measure defined in that methodology.

In clustering algorithms, each observation x_i is assigned to one and only one cluster k , where k is specified and less than N (total number of observations). This cluster assignment can be defined as $c(i) = k$ where $k \in \{1, \dots, K\}$. In this way, we will have the global dissimilarity of all clusters as

$$W(c) = \frac{1}{2} \sum_{k=1}^K \sum_{c(i)=k} \sum_{c(i')=k} D(x_i, x_{i'}) \quad (42.82)$$

where the inner summations give the within cluster dissimilarity or scatter. We can show that minimizing the within cluster scatter is equivalent to maximizing between cluster scatter, if we consider T as all possible cluster assignments and $B(c)$ as between clusters point scatter:

$$\begin{aligned} T &= \frac{1}{2} \sum_{i=1}^p \sum_{i'=1}^p D(x_i, x_{i'}) \\ &= \frac{1}{2} \sum_{k=1}^K \sum_{c(i)=k} \left[\sum_{c(i')=k} D(x_i, x_{i'}) + \sum_{c(i') \neq k} D(x_i, x_{i'}) \right] \\ &= W(c) + B(c) \end{aligned} \quad (42.83)$$

$$W(c) = T - B(c) \quad (42.84)$$

The best cluster assignment is $\hat{C} = \operatorname{argmin}_c W(c) = \operatorname{argmax}_c B(c)$. We can see that the number of cluster assignments needed to be checked is huge.

K-means Clustering

K-mean is the most popular iterative clustering method if all variables are quantitative type. In K-mean, squared Euclidean distance is used as the dissimilarity measure. In K-means, we initialize the algorithm so that for a given cluster assignment c , the total cluster variance is minimized with respect to the means of currently assigned clusters, m_1, \dots, m_k .

$$\begin{aligned} W(c) &= \frac{1}{2} \sum_{k=1}^K \sum_{c(i)=k} \sum_{c(i')=k} D(x_i, x_{i'}) \\ &= \sum_{k=1}^K \sum_{j=1}^p \hat{\operatorname{Var}}(x_j | c(i) = k) \times N_k^2 \end{aligned} \quad (42.85)$$

where

$$\hat{\operatorname{Var}}(x_j | c(i) = k) = \frac{1}{N_k} \sum_{c(i)=k} (x_{ij} - \bar{m}_{jk})^2 \quad (42.86)$$

$$\bar{m}_{jk} = \bar{x}_{jk} = \frac{1}{N_k} \sum_{c(i)=k} x_{ij} \quad (42.87)$$

N_k is the number of observations in cluster k . Considering the definitions above, $W(c)$ becomes:

$$W(c) = \min_{m_1, \dots, m_k} \sum_{k=1}^K \sum_{j=1}^p N_k \sum_{c(i)=k} (x_{ij} - \bar{m}_{jk})^2 \quad (42.88)$$

$\hat{C} = \min_{c, M} W(c, M)$ where M is the vector of means $M = \{m_1, \dots, m_k\}$. Now given a current set of means vector, we minimize $W(c, M)$ by assigning each observation to the closest current cluster mean, i.e., $c(i) = \min_{1 \leq k \leq K} \sum_{j=1}^p (x_{ij} - m_{jk})^2$ and repeat the process until cluster assignments do not change. There are many clustering algorithms based on variations of k -means modifications and centroid-based such as k -medoids, k -medians, or fuzzy c -means [33–35].

Hierarchical Clustering

One of the disadvantages of K -means or K -medoids is that the number of clusters and the initial assignments must be selected. Therefore, the clustering results depend on the analyst's choice for the number of clusters and a starting initialization. However, in hierarchical method based on the measure of dissimilarity given, we can find the agglomerative (bottom-up) or divisive (top-down) clusters. Agglomerative method starts from the bottom and at each level joins a pair of clusters into a single cluster based on the similarity measure. Divisive method performs the opposite by starting at the top and splitting each cluster into two clusters. In these methods, we get a tree-structured map of the clusters and dissimilarities between groups. Mostly, agglomerative is popular among analysts because of an interesting feature in its clustering map's graphical display called a dendrogram. The dendrogram is plotted in a way that the length (tree height) between each main cluster and subcluster is proportional to the dissimilarity between them. The terminal nodes of the dendrogram are the individual observations. In agglomerative method, we can perform single linkage, complete linkage, and group average clustering.

Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

DBSCAN is a popular density-based clustering algorithm proposed by Ester et al. in 1996 [36]. It is a nonparametric clustering algorithm that finds neighborhoods with high densities. How it differs from k -mean algorithm is that a constant ε is predetermined so that any two points with distance between them lower or equal to "eps" are considered as neighbors. Furthermore, the minimum number of data points within each neighborhood ε is also defined. DBSCAN defines the neighborhoods recursively and setting all other points outside of neighborhoods as Noise or outlier. Consequently, this algorithm works well with noisy datasets.

Dimension Reduction Techniques

Principal Components Analysis

Principal Components Analysis (PCA) is a useful method in many problems. The most important task of PCA is to repre-

sent the variability of a large number of correlated variables in a much lower dimension. PCA provides a tool that finds a low-dimensional representation of a data set, called principal components Z_1, \dots, Z_k , with as much information and variation as possible. In order to find the principal component of the features x_1, \dots, x_p , we look for normalized linear combinations of the features that have the largest variance.

$$Z_1 = \varnothing_{11}x_1 + \varnothing_{21}x_2 + \dots + \varnothing_{p1}x_p \quad (42.89)$$

where $\sum_{j=1}^p \varnothing_{j1}^2 = 1$, Z_1 is the first PC, and vector \varnothing_i is the PC loading vector. We need to solve the following optimization to find the maximum sample variance of z_{i1} which can be solved with eigen decomposition method.

$$\max_{\varnothing_{11}, \dots, \varnothing_{p1}} \left[\frac{1}{n} \sum_{i=1}^n \left(\sum_{j=1}^p \varnothing_{j1}x_{ij} \right)^2 \right] = \max \frac{1}{n} \sum_{i=1}^n z_{i1}^2 \quad (42.90)$$

$$\text{subject to } \sum_{j=1}^p \varnothing_{j1}^2 = 1 \quad (42.91)$$

The second principal component is again the linear combination of x_1, \dots, x_p with maximum variance that is uncorrelated with Z_1 . In order to find Z_2 , the same optimization problem should be solved with an additional constraint that the direction of \varnothing_2 should be orthogonal to the direction \varnothing_1 in order for them to be uncorrelated. It is recommended to scale all variables to have mean zero before performing PCA. One important metric in PCA is the proportion of variance explained (PVE) by each principal component. The cumulative PVE of the first M principal components can be calculated by summing over each of the first M PVEs.

$$\text{PVE}_m = \frac{\sum_{i=1}^n \left(\sum_{j=1}^p \varnothing_{jm}x_{ij} \right)^2}{\sum_{j=1}^p \sum_{i=1}^n x_{ij}^2} \quad (42.92)$$

Furthermore, PCA transformation can also be helpful as a preprocessing step before clustering the data. PCA helps identify those data points which bring higher variance in the original data and also removing multicollinearity among different features. The results of PCA depend on the scale of the variables, so it is better to scale each feature by its standard deviation.

PCA can also be used in regression problems. To perform principal components regression (PCR), we simply use principal components as predictors in a regression model instead of the original variables. Partial least squares (PLS) method works the same way as PCR [37, 38].

t-Distributed Stochastic Neighbor Embedding (t-SNE) unlike PCA is a nonlinear dimensionality reduction technique that is primarily used for visualization purposes of high-dimensional datasets. t-SNE is a newer machine learning algorithm compared to PCA and PLS developed in 2008 [39]. PCA works with quantitative variables; however, multiple correspondence analysis (MCA) is the PCA version for qualitative variables [40].

42.3.3 Semisupervised Learning

Semisupervised learning comes when the cost of labeling process may be large, some parts of labels are missing, or a fully labeled data is infeasible. In these situations, semisupervised learning can be used where we have a mixed dataset of labeled and unlabeled observations. All semisupervised algorithms assume the following about the data:

Continuity Assumption: Points which are closer to each other with high probability will have the same output label. Cluster Assumption: Assuming the data is divided into different clusters, points in the same cluster are more likely to have the same output label. Manifold Assumption: The data lie approximately on a manifold of much lower dimension than the input space. Semisupervised learning is rather new and evolving in machine learning. There are some popular algorithms such as self-training, generative methods, mixture models, graph-based methods, and semisupervised SVM (S3VMs) [41–47].

42.4 Case Study

Datasets containing energy consumption in Netherlands between 2010 and 2020 have been used to investigate supervised prediction techniques mentioned in previous sections [48]. Enexis, Liander, and Stedin are the three major energy providers of the Netherlands, and the yearly energy consumption of the areas under their administration is available online. Each year dataset consists of annual energy consumption of different neighborhood (including zip codes and street names) in different cities under the company's territory. Each dataset is for a specific network administrator for a specific year and contains the following columns: network manager, purchase area, street name, zip code from and zip code to (two columns for the range of zip codes covered), city name, number of connections (number of connections in that range of zip codes), delivery percentage (percentage of the net consumption of electricity or gas – the lower this number the more energy was given back to the grid for instance if that neighborhood had solar panels to produce energy), percentage of active connections, type of connection (principal type of connection in the zip code range), type of

Table 42.1 Summary of statistics

	Number of connections	Delivery %	% of active connections	Type connection %	Annual consumption low tariff %	Smart meter %	Year	Annual consumption
Count	156,791	156,791	156,791	156,791	156,791	156,791	156,791	156,791
Mean	30.17	99.28	91.45	77.28	25.93	20.40	2014.59	3478.69
Std	22.81	3.69	17.47	20.33	30.36	30.73	2.88	3220.31
Min	10	3.26	0	18	0	0	2010	52
25%	18	100	91.43	60	3.33	0	2012	1934
50%	25	100	97.67	82	11.54	2.5	2015	2509
75%	34	100	100	96	42.11	32	2017	3675
Max	669	100	100	100	100	100	2019	59,207

connection percentage, annual consume low tariff percentage (percentage of consume during the low tariff hours -from 10 p.m. to 7 a.m. and during weekends), smart meter percentage and finally annual consume kWh.

One of the important factors about this dataset is the information it has about Smart meters. Smart meters record energy consumption and send the observed data to the electricity supplier. This information can be used for by the suppliers for many applications such as monitoring and billing purposes. Since smart meters enable the intelligent controlling systems, they play an important role in the future energy grids. The most important benefits of smart meters are increasing the energy efficiency, reducing energy costs by enhancing the provision of ancillary services, and empowering utilities electricity production by integrating energy storages and renewable energies during high demands [49, 50]. Moreover, smart meters provide data for analysis of power quality and power outages. Smart meters provide a two-way communication between grid and smart meter which has data storage capability of outage records that can be used for predicting blackouts [51].

Moreover, the consumers can also get direct information about their energy consumption which motivates higher energy savings or consuming during low tariff times. The last remark gives us the second most important information in our dataset which is “percentage of consume during low tariff hour.” Furthermore, The Netherlands has aimed to ban petrol and diesel cars in the coming years making electric vehicles a sustainable transport alternative. Concerning the remarks mentioned about electricity in Netherlands, we chose this dataset to help enlighten some useful information.

42.4.1 Exploratory Data Analysis (EDA)

For the purpose of this chapter, we focus only on the dataset for the city of Amsterdam and predicting the coming year’s energy consumption based on the information available. Liander is the main company providing energy in Amsterdam. The final dataset consists of both categorical

and numerical features. Combining the datasets for all years, the final dataset consists of nine features or independent variables including “year” as one of the features and one response variable or dependent variable “energy consumption.” The goal in this case study would be to predict the energy consumption for the coming year of 2020.

The training data did not have any missing values to be omitted or imputed, but categorical variables with categories less than 10 times frequency have been omitted. Table 42.1 shows the summary of statistics for the numerical variables.

The histograms and pairwise scatterplot of numerical variables are presented below (Figs. 42.1 and 42.2 respectively). We can visually see the relationship between annual consumption versus all numerical variables (Fig. 42.2).

Moreover, the bar chart of the categorical variables (type of connections and network manager) versus the annual consumption is presented below (Fig. 42.3).

In order to evaluate the dependency among independent variables and assess multicollinearity, the correlation matrix is also presented (Fig. 42.4). We can see that annual consumption has a positive correlation with annual consumption of low-tariff electricity which is reasonable. Furthermore, we can see as years have passed the number of installed smart meters has increased, and consequently the percentage of low tariff annual consumption has risen significantly. This shows the importance of smart meters in increasing low-tariff consumption for bill management and decrease of annual consumption.

42.4.2 Modeling Selection Criteria

In order to evaluate the algorithms mentioned earlier, there are numbers of evaluation metrics. MSE (mean squared error) is the most basic and famous metric used for regression purposes. In order to compare different models, test MSE may not be enough since in all regression models we fit a model using least squares. More specifically, we estimate the regression coefficients such that residual sum of squares (RSS) is minimized. Therefore, other probabilistic statistical metrics that quantify both the model performance and the

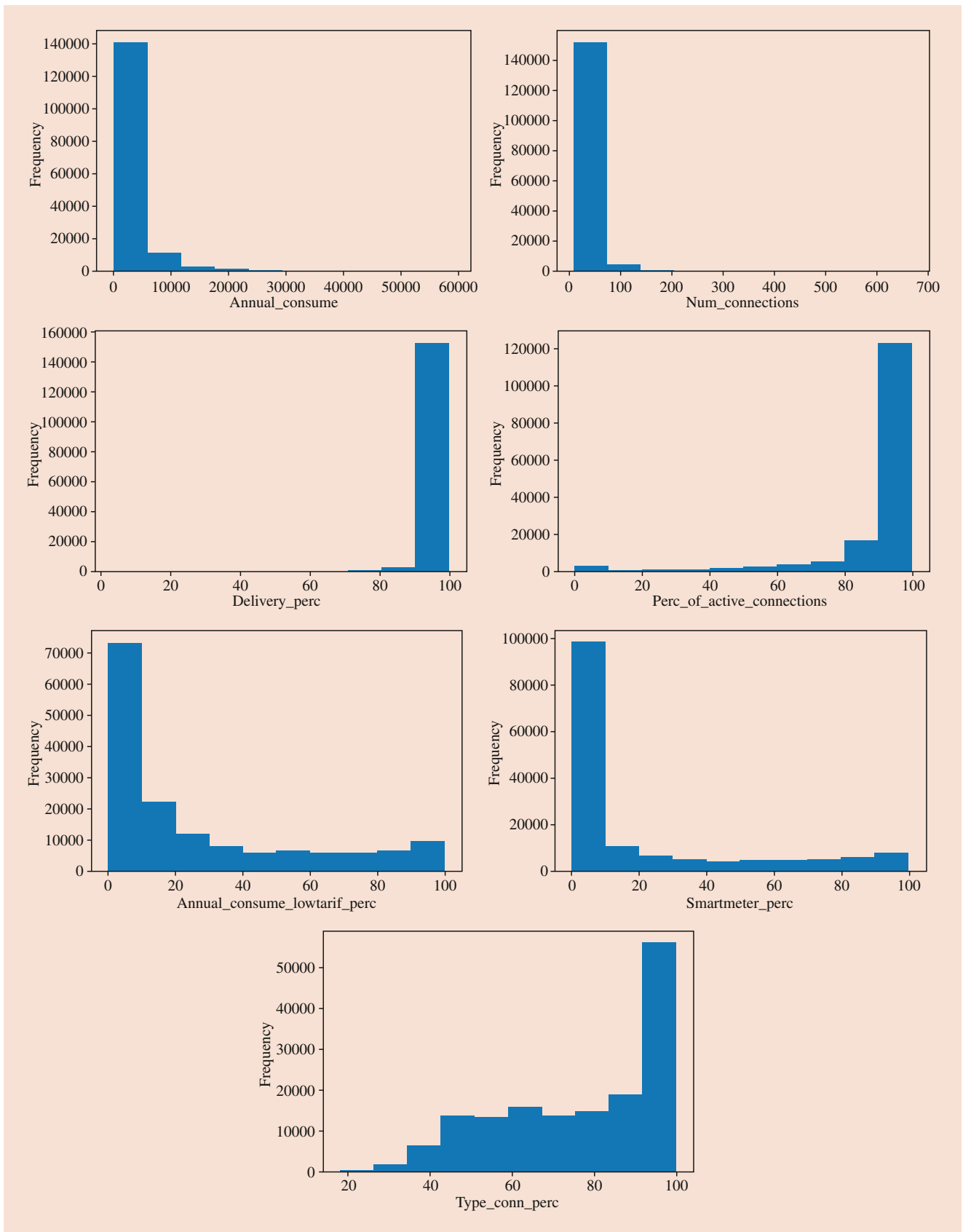


Fig. 42.1 Histograms of the numerical variables

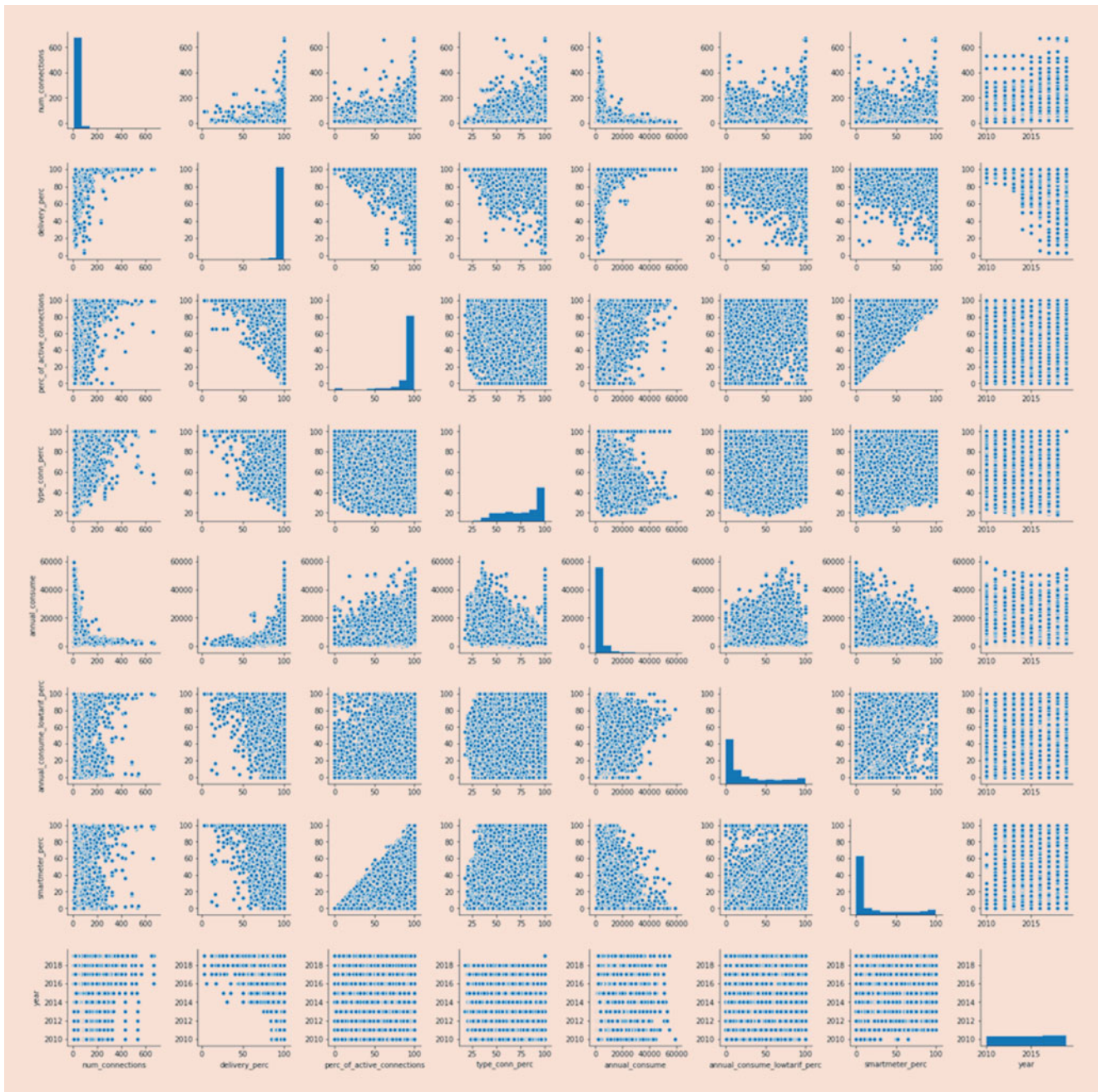


Fig. 42.2 Pair plot of numerical variables

complexity of the model are also investigated. Among some of most important ones are RMSE, MAE, R^2 , AIC, and BIC. Each of these metrics is insightful in different aspects. RMSE measures the difference between the model predictions and the truth values. A large RMSE is equivalent to a large average error, so smaller values of RMSE are preferable. RMSE presents the error in the units being measured which makes it easily interpretable to say how incorrect the model might be on unseen data. On the other hand, Mean Absolute Error (MAE) is more robust to outliers compared to RMSE which

punishes large errors. R^2 or the coefficient of determination is the proportion of the variance in the response variable that is explainable or predictable using the selected model and the independent variables in it.

Akaike Information Criterion (AIC) is derived from frequentist probability, and Bayesian Information Criterion (BIC) is derived from Bayesian probability – these are two ways of scoring models based on its complexity and goodness of the fit using loglikelihood. The lower AIC score is, the better the model is. For example, the AIC score is a trade-off

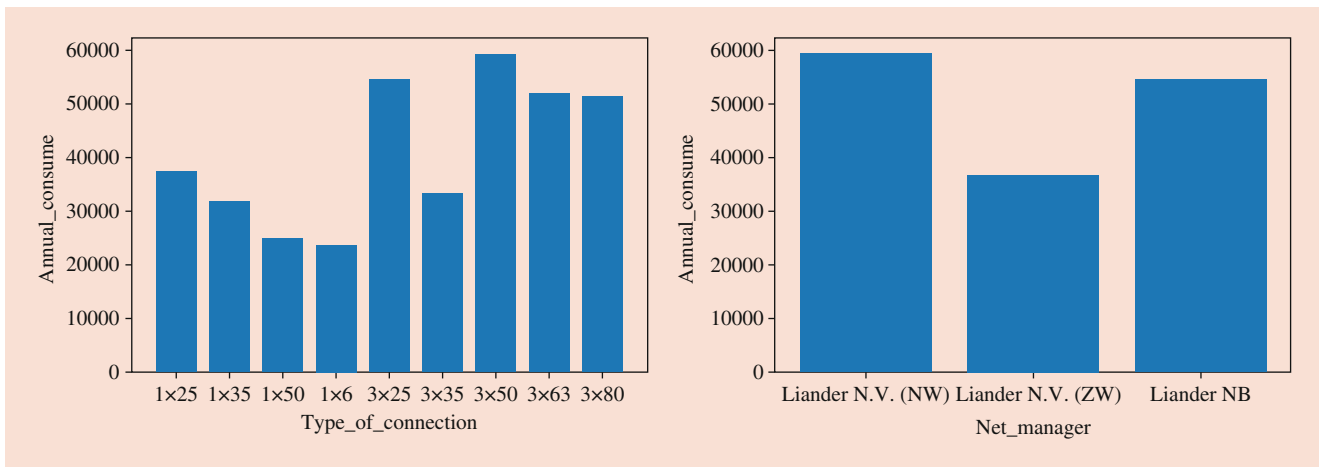


Fig. 42.3 Bar chart of the categorical variables versus annual consumption

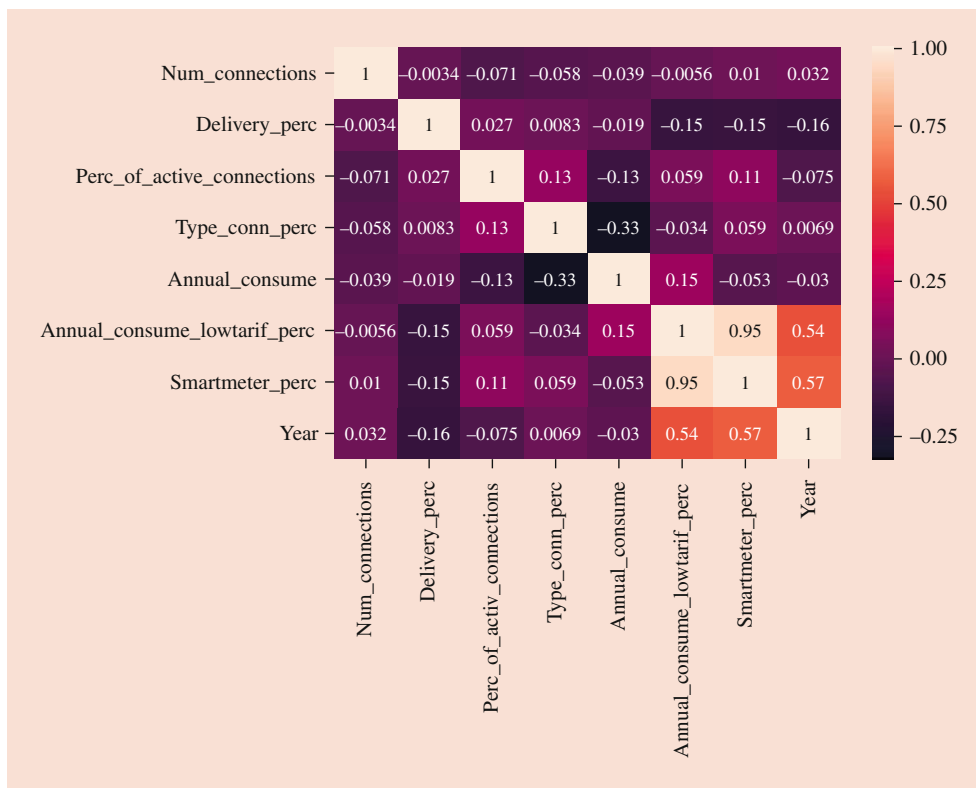


Fig. 42.4 Correlation matrix heatmap

between a high goodness-of-fit score and model simplicity, or the relative amount of information lost by a model and the risk of overfitting and underfitting in it.

Table 42.2 presents the definition of each metric. Here n is the number of observations, \hat{y}_i is the predicted value of observation i and y_i is the true value, k is the number of parameters in the model, and \hat{L} is the estimated maximum likelihood of the function for the model. We must note that another equivalent formulation for AIC and BIC when assuming a Gaussian model for the dataset with unknown

constant variance are $2k + n \ln \sum_{i=1}^n (y_i - \hat{y}_i)^2$ and $k \ln n + n \ln \sum_{i=1}^n (y_i - \hat{y}_i)^2$, respectively.

42.4.3 Results Analysis

The study dataset for Amsterdam energy consumption consists of 156,963 observations. Total 30% of the dataset is set as a testing set, and the remaining 70% is used for training

and building models. After exploratory data analysis, we used Cross-Validation (CV) to test the generalization ability of the models and also tuning parameters in different models. In this machine learning study, after the data is split into training and testing set, 10-Fold CV is implemented in which we further split our training set into ten numbers of subsets (folds).

Each time, we fit the model on nine ($K - 1$) of the folds and test it on the tenth (K th) fold. After finding the best parameters of the models using K -Fold CV and hyper parameter tuning, the final models are used to predict the test dataset and the results are presented in Table 42.3.

Here Ensemble models such as boosting and bagging algorithms have been used with decision tree as the base learner of the model. Moreover, we can see that more advanced machine learning models such as XGBoosting, LightGBM, or Neural Network perform better compared to simpler one such as OLS, Ridge, or KNN for our dataset. For instance, R -square increases from 0.5 for linear model to 0.7 for more flexible. More precisely, LightGBM outperforms all algorithms with the lowest MSE and RMSE and highest R^2 . XGboosting comes in the second-best model in terms of MSE and RMSE. However, if we compare MAE metric which is a more robust metric, we can see that AdaBoost has the lowest

MAE, and afterward XGBoosting and Gradient Boosting methods come as second-best models.

Lowest AIC and BIC again present lightGBM as the best model. In this study, we have implemented a simple sequential neural network model with three hidden layers using Keras. Keras is an open source deep learning library that we have the ability to define models as a sequence of layers (Sequential model) and add layers to add more complexity to the model [52]. As discussed previously, Neural Networks might not perform very well for tabular structured datasets which is the case for our dataset too. Overall, in our case lightGBM outperforms all other algorithms because of its flexibility, and AdaBoost is the more robust model to choose. XGboosting comes in second place for almost all metrics which makes it a good choice too, but this is still subjective to the analysts and their objectives.

The main purpose in developing models in ML is to use them as a prediction tool in unseen data and provide useful information for our problem. Considering that, we used the model with the better accuracy overall, LightGBM, to predict the year 2020 energy consumption in Amsterdam. Performing prediction for year 2020 with the fitted LightGBM model on years 2010–2019, we get the mean absolute percentage deviation (MAPD) or the relative error of 28.17% which is rather a good result in real-world problems. In other words, Amsterdam energy consumption can be predicted using the ML algorithms with 28.17% error rate.

Table 42.2 Model evaluation metrics

Metric	Definition
MSE	$\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$
RMSE	$\sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$
MAE	$\frac{\sum_{i=1}^n y_i - \hat{y}_i }{n}$
R^2	$1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$
AIC	$2k - 2 \ln(\hat{L})$
BIC	$k \ln(n) - 2 \ln(\hat{L})$

Table 42.3 The summary table of algorithms results

Metric	MSE	RMSE	MAE	R^2	AIC	BIC
Model						
OLS regression	5242734.080	2289.702	1332.860	0.499	727826.573	727992.988
Lasso	5242755.679	2289.706	1332.875	0.499	727826.767	727993.182
Ridge regression	5242738.321	2289.703	1332.860	0.499	727826.611	727993.026
Elastic net	5246414.978	2290.505	1333.276	0.499	727859.586	728026.002
K -nearest neighbors	4271067.914	2066.656	1122.253	0.592	718184.759	718351.175
Linear SVR	9155695.147	3025.838	1414.157	0.125	754051.809	754218.224
Nonlinear SVR	9575522.378	3094.434	1427.333	0.085	756160.710	756327.125
Decision tree	4331435.097	2081.210	1129.735	0.586	718844.939	719011.354
Bagging	3032726.834	1741.473	958.436	0.710	702078.867	702245.282
Random forest	3039058.614	1743.290	961.350	0.710	702176.971	702343.387
AdaBoost	3082160.014	1755.608	939.577	0.705	702839.400	703005.816
Gradient boosting	3117359.154	1765.604	944.530	0.702	703373.543	703539.959
XGBoost	2981607.579	1726.733	942.577	0.715	701279.242	701445.657
LightGBM	2957935.150	1719.865	963.400	0.717	700904.294	701070.710
Neural network	4197413.977	2048.759	1114.023	0.599	717366.520	717532.936

42.5 Summary

For decades, scientists in machine learning have tried to explore and discover meaningful patterns from the raw data. Machine learning has become a new and interesting topic that is defined as linear and nonlinear processing algorithms in or-

der to learn underlying representation of datasets. Therefore, it has provided new opportunities and solutions to many real-world problems and created many applications. This chapter surveys the state-of-the-art algorithms and techniques in machine learning starting with a history of machine learning since the 1950s and moves to the recent and popular machine learning algorithms. The most popular categorizations of the algorithms such as learning types are discussed to present differences between these algorithms and the necessity on developing new novel algorithms. A little introduction to deep learning as a black box solution for many applications that can work with massive amount of unstructured data and its difference with Machine learning is discussed.

Then, the popular and key algorithms as well as techniques in this are presented with detailed mathematical formulations for better understanding. Several findings of this chapter are summarized in this section. First, traditional classification and regression methods and several supervised machine learning algorithms are introduced each of which have assumptions, advantages, and limits. For instance, KNN is a very flexible nonparametric algorithm which is simple to implement and fits nonlinear boundary/predictions to datasets. KNN is rather robust to noise and outliers in the input data, however not very efficient compared to other models since the entire dataset is used for predicting each new observation. SVM works well with complex and high-dimensional dataset specially because of kernel trick and tends to avoid overfitting because of regularization. But the model might underperform for large datasets which also has long training time. Optimizing the kernel function, lack of interpretability of the model, and variables importance and weights are other limits of SVM. Linear and Quadratic Discriminant Analysis models assume normality for the conditional densities which might not be true in many applications. Logistic regression, on the other hand, assumes linearity of independent variables and log odds of the dependent variable which can be a limit. Moreover, logistic regression returns the probabilities for only two possible outcomes in a classification problem. On the other hand, Naive Bayes classifier performs well in multiclass prediction but has the assumption of independency between predictors which is impossible in many real-life problems. Classification and Regression Tree (CART) is simple to implement and interpretable in business purposes but has limits such as sensitivity to even small changes in the data, high training time, and low accuracy. Traditional regression methods with regularizations have been discussed in detail and show how they overcome overfitting in regression problems. Ridge regression, for example, performs L2 regularization and does not produce unbiased estimators, so the variance of the model can be very large. Moreover, Ridge model does not perform feature selection whereas Lasso with L1 regularization does so.

Afterward, more advanced algorithms such as ensemble models and boosting algorithms are discussed. Bagging basi-

cally consists of weak learners operating independently from each other in parallel and average them at the end. Boosting, however, considers weak learners sequentially operating in such a way that each model depends on the previous ones, trying to improve it, and combines them by averaging all at the end. Popular Boosting algorithms such as AdaBoost, XGBoost, and lightGBM are explained thoroughly. The differences between each of the algorithms and how they built up upon each other show the historical time line. Different types of Neural Networks and their applications have been discussed.

In the following section, this chapter reviews the most successful unsupervised learning methods such as clustering analysis and dimension reduction techniques. K-Means is most relatively simple to implement and computationally faster than hierarchical clustering. However, in K-Means the number of clusters should be chosen beforehand. Hierarchical clustering with structured tree diagram for the number of clusters is more informative but has time complexity and is sensitive to outliers. PCA as a statistical technique to reduce the number of features (dimension) of data is discussed. Moreover, definition and challenges in semisupervised learning have also been briefly provided.

Thereafter, a small case study is presented to illustrate some of the algorithms and their differences more explicitly. To summarize, this chapter discusses many algorithms and challenges in machine learning and provides several existing solutions to these challenges. However, there are still challenges and possible future research venues that need to be addressed in the future of machine learning. Some of which that have been found in this chapter are as follows: The interpretability of many of machine-learning algorithms can be investigated in the future. Many machine learning models need extensive datasets for training before predicting the unseen data, and this can be a problem when small datasets are only available, or the data needs to be processed in real time. In this regard, small sample learning, one-shot learning and zero-shot learning, and online learning have got attention among researchers the past years. Additionally, the machine learning industry is gradually shifting its focus to unsupervised and semisupervised learning instead of supervised learning so that the machine can process data without manual human labels. All in all, machine learning as a new fast evolving area has numerous challenges as well as opportunities in a variety of applications.

References

1. Samuel, A.L.: Some studies in machine learning using the game of checkers. *IBM J. Res. Dev.* **3**(3), 210–229 (1959)
2. Rosenblatt, F.: *The Perceptron, a Perceiving and Recognizing Automaton Project Para.* Cornell Aeronautical Laboratory, Cornell University, Ithaca, NY, 85–460 (1957)
3. Linnainmaa, S.: Taylor expansion of the accumulated rounding error. *BIT Numer. Math.* **16**(2), 146–160 (1976)

4. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Internal Representations by Error Propagation. California Univ San Diego La Jolla Inst for Cognitive Science (1985)
5. Cover, T., Hart, P.: Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory*. **13**(1), 21–27 (1967)
6. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)
7. Ho T.K.: Random decision forests. In: Proceedings of 3rd international conference on document analysis and recognition, vol. 1, pp. 278–282 (1995)
8. Klinger, C., Landeg, O., Murray, V.: Power outages, extreme events and health: a systematic review of the literature from 2011–2012. *PLoS Curr.* **1** (2014)
9. Mnih, V. et al.: Playing atari with deep reinforcement learning. *arXiv Prepr. arXiv1312.5602* (2013)
10. Mnih, V., et al.: Human-level control through deep reinforcement learning. *Nature*. **518**(7540), 529–533 (2015)
11. Kaelbling, L.P., Littman, M.L., Moore, A.W.: Reinforcement learning: a survey. *J. Artif. Intell. Res.* **4**, 237–285 (1996)
12. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (2018)
13. Sutton, R.S., Barto, A.G.: Introduction to Reinforcement Learning, vol. 135. MIT Press, Cambridge (1998)
14. Hasselt, H.V.: Double Q-learning. *Adv. Neural Inf. Process. Syst.* **23**, 2613–2621 (2010)
15. Watkins, C.J.C.H., Dayan, P.: Q-learning. *Mach. Learn.* **8**(3–4), 279–292 (1992)
16. Schmidhuber, J.: Deep learning in neural networks: An overview. *Neural Netw.* **61**, 85–117 (2015)
17. Mikolov, T., Karafiát, M., Burget, L., Černocký, J., Khudanpur, S.: Recurrent neural network based language model. In: Eleventh annual conference of the international speech communication association (2010)
18. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) *Advances in Neural Information Processing Systems*, **60**(6), 84–90. Curran Associates, Inc. (2012)
19. Abdel-Hamid, O., Mohamed, A., Jiang, H., Deng, L., Penn, G., Yu, D.: Convolutional neural networks for speech recognition. *IEEE/ACM Trans. Audio, Speech, Lang. Process.* **22**(10), 1533–1545 (2014)
20. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.-A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on Machine learning, pp. 1096–1103 (2008)
21. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A.: Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* **11**(Dec), 3371–3408 (2010)
22. Bengio, Y.: Learning deep architectures for AI. *Found. Trends Mach. Learn.* **2**(1), 1–27 (2009)
23. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Ian claimed as feature learner. Improved techniques for training gans. *Adv. Neural Inf. Process. Syst.* **29**, 1–10 (2016)
24. Ng, A.Y., Jordan, M.I.: On discriminative vs. generative classifiers: a comparison of logistic regression and naive Bayes. *Neural. Process. Lett.* **28**(3), 169–187 (2008)
25. Quinlan, J.R.: Induction of decision trees. *Mach. Learn.* **1**(1), 81–106 (1986)
26. Salzberg, S.L.: C4.5: programs for machine learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993. *Mach. Learn.* **16**(3), 235–240 (1994)
27. Ke, G., et al.: LightGBM: A highly efficient gradient boosting decision tree. *Adv. Neural Inf. Process. Syst.* **2017**(Nips), 3147–3155 (2017)
28. Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A.V., Gulin, A.: Catboost: unbiased boosting with categorical features. *Adv. Neural Inf. Process. Syst.* **2018**(Section 4), 6638–6648 (2018)
29. Haykin, S.: *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR (1994)
30. Rosenblatt, F.: The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **65**(6), 386 (1958)
31. Bishop, C.M.: *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford (1995)
32. Paliwal, M., Kumar, U.A.: Neural networks and statistical techniques: a review of applications. *Expert Syst. Appl.* **36**(1), 2–17 (2009)
33. Park, H.-S., Jun, C.-H.: A simple and fast algorithm for K-medoids clustering. *Expert Syst. Appl.* **36**(2), 3336–3341 (2009)
34. Arora, S., Raghavan, P., Rao, S.: Approximation schemes for Euclidean k-medians and related problems. In: Proceedings of the thirtieth annual ACM symposium on Theory of computing, pp. 106–113 (1998)
35. Bezdek, J.C., Ehrlich, R., Full, W.: FCM: the fuzzy c-means clustering algorithm. *Comput. Geosci.* **10**(2–3), 191–203 (1984)
36. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. *Kdd.* **96**(34), 226–231 (1996)
37. Wold, S., Sjöström, M., Eriksson, L.: PLS-regression: a basic tool of chemometrics. *Chemom. Intell. Lab. Syst.* **58**(2), 109–130 (2001)
38. Jolliffe, I.T.: A note on the use of principal components in regression. *J. R. Stat. Soc. Ser. C. Appl. Stat.* **31**(3), 300–303 (1982)
39. van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**(Nov), 2579–2605 (2008)
40. Abdi, H., Valentin, D.: Multiple correspondence analysis. *Encycl. Meas. Stat.* **2**, 651–666 (2007)
41. Triguero, I., García, S., Herrera, F.: Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowl. Inf. Syst.* **42**(2), 245–284 (2015)
42. Fazakis, N., Karlos, S., Kotsiantis, S., Sgarbas, K.: Self-trained LMT for semisupervised learning. *Comput. Intell. Neurosci.* **2016**, 1–13 (2016)
43. Zhu, X., Goldberg, A.B.: Introduction to semi-supervised learning. *Synth. Lect. Artif. Intell. Mach. Learn.* **3**(1), 1–130 (2009)
44. Chapelle, O., Scholkopf, B., Zien, A.: *Semi-supervised learning* (Chapelle, O. et al., eds.; 2006) [book reviews]. *IEEE Trans. Neural Netw.* **20**(3), 542 (2009)
45. Zhu, X.J.: *Semi-Supervised Learning Literature Survey*. University of Wisconsin-Madison, Department of Computer Sciences (2005)
46. Kingma, D.P., Mohamed, S., Rezende, D.J., Welling, M.: Semi-supervised learning with deep generative models. In: *Advances in neural information processing systems*, pp. 3581–3589 (2014)
47. Zhu, X., Ghahramani, Z., Lafferty, J.D.: Semi-supervised learning using gaussian fields and harmonic functions. In: Proceedings of the 20th International conference on Machine learning (ICML-03), pp. 912–919 (2003)
48. <https://978-1-4471-7503-2/www.kaggle.com/lucabasa/dutch-energy>
49. Mahani, K., Jamali, M.A., Nazemi, S.D., Jafari, M.A.: Economic and operational evaluation of PV and CHP combined with energy storage systems considering energy and regulation markets. In: 2020 IEEE Texas Power and Energy Conference (TPEC), pp. 1–6 (2020)
50. Mahani, K., Nazemi, S.D., Jamali, M.A., Jafari, M.A.: Evaluation of the behind-the-meter benefits of energy storage systems with consideration of ancillary market opportunities. *Electr. J.* **33**(2), 106707 (2020)

51. Arabzadeh Jamali, M.: Study of Power Recoverability through Optimal Design of Energy Storage Systems. Rutgers University-School of Graduate Studies (2019)
52. Keras 2.3.1. [Online]. Available: <https://github.com/keras-team/keras/releases/tag/2.3.1>



Maryam Arabzadeh Jamali is currently a PhD candidate in the Department of Industrial and Systems Engineering at Rutgers University. She received her MS degree in Industrial and Systems Engineering from Rutgers University and BS degree in Industrial Engineering, from Sharif University of Technology, Iran. Her research focuses on the reliability and maintainability of complex systems and their interplay with AI and machine learning models.



Hoang Pham is a Distinguished Professor and former Chairman of the Department of Industrial & Systems Engineering at Rutgers University. He is the author or coauthor of 7 books and has published over 200 journal articles, 100 conference papers, and edited 20 books. His numerous awards include the 2009 IEEE Reliability Society *Engineer of the Year Award*. He is a Fellow of the IEEE and IISE.



Covariance Estimation via the Modified Cholesky Decomposition

Xiaoning Kang, Zhiyang Zhang, and Xinwei Deng

Contents

43.1	Introduction	887
43.2	Review of Modified Cholesky Decomposition	888
43.2.1	MCD for Precision Matrix Estimation.....	888
43.2.2	MCD for Covariance Matrix Estimation.....	889
43.2.3	MCD for Banded Matrix Estimation.....	890
43.2.4	Adaptive Banding in the MCD.....	890
43.3	Ordering Issue in the MCD	892
43.4	Real Applications	893
43.4.1	MCD for Varying Covariance Matrix Estimation in Multivariate Time Series.....	893
43.4.2	MCD for Portfolio Optimization.....	894
43.4.3	MCD for Linear Discriminant Analysis.....	895
43.5	Numerical Study	896
43.6	Discussion	897
Appendix		897
References		899

implies certain sparse structure on the covariance and precision matrices. In this chapter, we first overview the Cholesky-based covariance and precision matrices estimation. It is known that the Cholesky-based matrix estimation depends on a prespecified ordering of variables, which is often not available in practice. To address this issue, we then introduce several techniques to enhance the Cholesky-based estimation of covariance and precision matrices. These approaches are able to ensure the positive definiteness of the matrix estimate and applicable in general situations without specifying the ordering of variables. The advantage of Cholesky-based estimation is illustrated by numerical studies and several real-case applications.

Keywords

Banded matrix · Cholesky-based · Linear discriminant analysis · Ordering of variables · Positive definite · Portfolio optimization · Sparsity

Abstract

In many engineering applications, estimation of covariance and precision matrices is of great importance, helping researchers understand the dependency and conditional dependency between variables of interest. Among various matrix estimation methods, the modified Cholesky decomposition is a commonly used technique. It has the advantage of transforming the matrix estimation task into solving a sequence of regression models. Moreover, the sparsity on the regression coefficients

43.1 Introduction

Estimation of covariance and precision matrices is of fundamental importance in the multivariate analysis [5]. It has received wide attentions of scholars in various engineering applications such as additive manufacturing [30, 31], biomedical engineering [17], and tissue engineering [39]. The resultant matrix estimation has also been widely used in various statistical methods and applications. For example, dimension reduction via the principal component analysis (PCA) usually relies on the estimation of covariance matrix. In the classification problem, the linear discriminant analysis (LDA) constructs the classification rule through the precision matrix. In the financial area, the portfolio optimization uses the precision matrix to minimize the portfolio risk. In signal

X. Kang
International Business College and Institute of Supply Chain Analytics,
Dongbei University of Finance and Economics, Dalian, China
e-mail: kangxiaoning@dufe.edu.cn; xiaoningmike@126.com

Z. Zhang · X. Deng (✉)
Department of Statistics, Virginia Tech, Blacksburg, VA, USA
e-mail: zhiyangz@vt.edu; xdeng@vt.edu

processing, the covariance matrix helps to distinguish between signals and noise. The covariance and precision matrices also arise in the graphical models, multivariate volatility, weather forecasting, social network, fMRI analysis, and so forth.

For the matrix estimation, a desirable property is the sparsity in the sense that some elements in an estimated matrix are zeros [13, 16, 20, 26, 37]. A sparse covariance matrix estimate is useful for the subsequent statistical analysis, such as inferring the correlation pattern among the predictor variables. A sparse precision matrix estimate often implies the conditional independence among the corresponding predictor variables. Therefore, a variety of classical approaches has been developed in literature for estimating the covariance and precision matrices with particular interest of sparse structure. Yuan and Lin [38] introduced the graphical Lasso (Glasso) model, which gives a sparse precision matrix estimate by imposing an L_1 penalty on the negative log-likelihood function. Bickel and Levina [1] proposed to threshold the small elements of the sample covariance matrix directly to zeroes with a large number of predictor variables. More work on the sparse estimation of the covariance and precision matrices can be found in [3–5, 10, 15, 21, 27, 32, 34, 36], among others.

Another important property of covariance and precision matrices is that they need to be positive definite for proper inference. The modified Cholesky decomposition (MCD) is a popular and commonly used technique for the matrix estimation, which ensures the estimated matrix to be positive definite. The MCD provides an unconstrained and statistically interpretable parameterization of a matrix by sequentially regressing the variables in a random variable vector. This method reduces the challenge of estimating a covariance or precision matrix into an easier task of solving a sequence of linear regression models. However, it is known that the Cholesky-based matrix estimation relies on the preknowledge of the ordering of variables. When the variable ordering is not naturally available, we need to consider several techniques to enhance the Cholesky-based estimation of covariance and precision matrices. Such techniques are able to ensure the positive definiteness of the matrix estimate and applicable in general situations without specifying the ordering of variables.

The remaining of this chapter is organized as follows. Sect. 43.2 provides a comprehensive overview of the MCD for the estimation of covariance and precision matrices, respectively. We then point out the variable ordering issue in the MCD and consider a couple of techniques to address the variable ordering issue in Sect. 43.3. Several real-data applications and numerical examples are presented in Sects. 43.4 and 43.5 to examine the performances of the Cholesky-based matrix estimates. We conclude this chapter with some discussion in Sect. 43.6.

43.2 Review of Modified Cholesky Decomposition

In this section, we review the modified Cholesky decomposition (MCD) in detail for estimating the covariance and precision matrices, respectively. As proposed by Pourahmadi [24], the MCD approach is statistically meaningful and guarantees the positive definiteness of a matrix estimate. The sparsity can be encouraged in the estimated matrix via the MCD technique.

43.2.1 MCD for Precision Matrix Estimation

Without loss of generality, suppose that $\mathbf{X} = (X_1, \dots, X_p)'$ is a p -dimensional random vector with mean $\mathbf{0}$ and covariance matrix Σ . Denote by $\mathbf{x}_1, \dots, \mathbf{x}_n$ the n independent and identically distributed observations following a multivariate distribution with mean $\mathbf{0}$ and covariance matrix Ω^{-1} , where $\Omega = \Sigma^{-1}$ is the precision matrix. The key idea of the MCD is that Ω can be diagonalized by a lower triangular matrix constructed from the regression coefficients when X_j is regressed on its predecessors X_1, \dots, X_{j-1} . Specifically, for $j = 2, \dots, p$, define

$$\begin{aligned} X_j &= \sum_{i=1}^{j-1} a_{ji} X_i + \epsilon_j \\ &= \mathbf{Z}_j^T \mathbf{a}_j + \epsilon_j, \end{aligned} \quad (43.1)$$

where $\mathbf{Z}_j = (X_1, \dots, X_{j-1})'$, and $\mathbf{a}_j = (a_{j1}, \dots, a_{jj-1})'$ is the corresponding vector of regression coefficients. The error term ϵ_j has population expectation $E(\epsilon_j) = 0$ and population variance $\text{Var}(\epsilon_j) = d_j^2$. Hence, a lower triangular matrix \mathbf{A} can be formed as

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 \\ a_{21} & 0 & 0 & \dots & 0 \\ a_{31} & a_{32} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{p1} & a_{p2} & \dots & a_{p,p-1} & 0 \end{pmatrix},$$

which contains all the regression coefficients in (43.1). Also define

$$d_j^2 = \text{Var}(\epsilon_j) = \begin{cases} \text{Var}(X_1), & j = 1, \\ \text{Var}(X_j - \mathbf{Z}_j^T \mathbf{a}_j), & j = 2, \dots, p. \end{cases} \quad (43.2)$$

Let $\mathbf{D} = \text{diag}(d_1^2, \dots, d_p^2)$ be the diagonal covariance matrix of the vector $\boldsymbol{\epsilon} = (\epsilon_1, \dots, \epsilon_p)'$. Then, move the term $\sum_{k=1}^{j-1} a_{jk} X_k$ in Eq. (43.1) to the left side and rewrite it in the following matrix form:

$$\boldsymbol{\epsilon} = (\mathbf{I} - \mathbf{A})\mathbf{X} = \mathbf{TX}, \quad (43.3)$$

where \mathbf{I} represents the $p \times p$ identity matrix and $\mathbf{T} = \mathbf{I} - \mathbf{A}$ is a unit lower triangular matrix having ones on its diagonal. The matrices \mathbf{T} and \mathbf{D} are called the Cholesky factor matrices. By taking the variance operator on both sides of Eq. (43.3), one can easily obtain

$$\mathbf{D} = \text{Var}(\boldsymbol{\epsilon}) = \text{Var}(\mathbf{TX}) = \mathbf{T}\boldsymbol{\Sigma}\mathbf{T}',$$

and thus

$$\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1} = \mathbf{T}'\mathbf{D}^{-1}\mathbf{T}. \quad (43.4)$$

As a result, the decomposition (43.1) converts the constraint entries of $\boldsymbol{\Sigma}$ into two groups of unconstrained “regression” and “variance” parameters. Conceptually, this approach reduces the challenge of modeling a precision matrix into the task of solving $(p - 1)$ linear regression models, which is much easier to implement.

A straightforward estimate $\hat{\mathbf{T}}$ of \mathbf{T} can be obtained from the least squares estimates of the regression coefficients

$$\hat{\mathbf{a}}_j = \arg \min_{\mathbf{a}_j} \|\mathbf{x}^{(j)} - \mathbb{Z}^{(j)}\mathbf{a}_j\|_2^2, \quad j = 2, \dots, p, \quad (43.5)$$

where $\mathbf{x}^{(j)}$ is the j th column of the data matrix $\mathbb{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)'$ and $\mathbb{Z}^{(j)} = (\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(j-1)})$ stands for the first $(j-1)$ columns of \mathbb{X} . The estimate $\hat{\mathbf{D}}$ of \mathbf{D} is constructed from the corresponding residual variances according to (43.2). Because the optimization (43.5) uses the ordinary least squares, this approach of precision matrix estimation is only suitable in low-dimensional settings with the number of predictor variables p smaller than the sample size n . When applying the MCD to the high-dimensional situations where p is close to or even larger than n , the least squares estimation is inaccurate or not available. In such cases, a natural idea is to employ the Lasso regularization [33] to shrink the regression estimates and encourage the sparsity on the Cholesky factor matrix \mathbf{T}

$$\hat{\mathbf{a}}_j = \arg \min_{\mathbf{a}_j} \|\mathbf{x}^{(j)} - \mathbb{Z}^{(j)}\mathbf{a}_j\|_2^2 + \lambda_j \|\mathbf{a}_j\|_1, \quad j = 2, \dots, p, \quad (43.6)$$

where $\lambda_j \geq 0$ is a tuning parameter and $\|\cdot\|_1$ stands for the vector L_1 norm. Note that the penalty in Eq. (43.6) is often suitable for data with large number of variables such as engineering data, social network data, and imaging data, since their underlying matrix is usually sparse with no specific sparse pattern. The optimization problem (43.6) can be solved by the coordinate descent algorithm [12]. The tuning parameters are determined by the cross-validation scheme

for each Lasso regression. In addition, one can alternatively consider the estimation of $\mathbf{a}_2, \dots, \mathbf{a}_p$ in the Cholesky factor matrix \mathbf{T} under a joint fashion as

$$\hat{\mathbf{a}}_2, \dots, \hat{\mathbf{a}}_p = \arg \min_{\mathbf{a}_2, \dots, \mathbf{a}_p} \sum_{j=2}^p \|\mathbf{x}^{(j)} - \mathbb{Z}^{(j)}\mathbf{a}_j\|_2^2 + \lambda \sum_{j=2}^p \|\mathbf{a}_j\|_1. \quad (43.7)$$

Such a joint estimation approach is used in [6, 13, 14, 40], among others.

After obtaining $\hat{\mathbf{a}}_j$, the lower triangular matrix $\hat{\mathbf{T}}$ is established with ones on its diagonal and $\hat{\mathbf{a}}_j'$ as its j th row. Meanwhile, the diagonal matrix $\hat{\mathbf{D}}$ has its j th diagonal element equal to \hat{d}_j^2 , where

$$\hat{d}_j^2 = \begin{cases} \widehat{\text{Var}}(\mathbf{x}^{(1)}), & j = 1, \\ \widehat{\text{Var}}(\mathbf{x}^{(j)} - \mathbb{Z}^{(j)}\hat{\mathbf{a}}_j), & j = 2, \dots, p, \end{cases}$$

where $\widehat{\text{Var}}(\cdot)$ denotes the sample variance. Consequently,

$$\hat{\boldsymbol{\Omega}} = \hat{\mathbf{T}}'\hat{\mathbf{D}}^{-1}\hat{\mathbf{T}} \quad (43.8)$$

is a sparse estimate for the precision matrix $\boldsymbol{\Omega}$.

Remark 1 For the optimization problem in (43.6) with $\lambda_j = 0$ and $n > p$, the estimated precision matrix in (43.8) is equivalent to the inverse of the sample covariance matrix.

43.2.2 MCD for Covariance Matrix Estimation

From (43.4), it is easy to see that a Cholesky-based estimate for a covariance matrix can be expressed as

$$\boldsymbol{\Sigma} = \mathbf{T}^{-1}\mathbf{D}\mathbf{T}'^{-1}. \quad (43.9)$$

The construction of the Cholesky factor matrices (\mathbf{T}, \mathbf{D}) through a series of linear regressions (43.1) is thoroughly discussed in Sect. 43.2.1. Such a covariance matrix estimate via the MCD can perform well if one does not require the estimated covariance matrix to be sparse. In other words, the estimate obtained based on Eq. (43.9) generally does not have sparse structure even though the Cholesky factor matrix \mathbf{T} is sparse. This is because the matrix \mathbf{T}^{-1} often does not inherit any sparse property from \mathbf{T} , leading to a dense estimate of covariance matrix $\boldsymbol{\Sigma}$. It is thus not convenient to impose a sparse structure on the estimate of $\boldsymbol{\Sigma}$ via Eq. (43.9).

Alternatively, one can consider a Cholesky-based latent variable regression model, which enables the regularization more easily. Write $\mathbf{X} = \mathbf{L}\boldsymbol{\epsilon}$, implying the predictor variable X_j is regressed on its previous latent variables $\epsilon_1, \dots, \epsilon_{j-1}$,

and hence $\mathbf{L} = (l_{ji})_{p \times p}$ is a unit lower triangular matrix constructed from the regression coefficients of the following sequential regressions:

$$X_j = \mathbf{l}_j^T \boldsymbol{\epsilon} = \sum_{i=1}^{j-1} l_{ji} \epsilon_i + \epsilon_j, \quad j = 2, \dots, p, \quad (43.10)$$

where $\mathbf{l}_j = (l_{ji})$ is the j th row of \mathbf{L} . Here $l_{jj} = 1$ and $l_{ji} = 0$ for $i > j$. This decomposition is interpreted as resulting from a different sequence of regressions, where each variable X_j is regressed on all the previous latent variable $\epsilon_1, \dots, \epsilon_{j-1}$ rather than themselves. As a result, this form of the MCD by the latent variable regression model provides a re-parameterization of the covariance matrix

$$\begin{aligned} \text{Var}(\mathbf{X}) &= \text{Var}(\mathbf{L}\boldsymbol{\epsilon}) \\ \boldsymbol{\Sigma} &= \mathbf{L}\mathbf{D}\mathbf{L}' \end{aligned}$$

This decomposition connects the covariance matrix $\boldsymbol{\Sigma}$ with linear regressions (43.10), such that the Lasso penalty can be imposed on the coefficients of the linear regressions, thus conveniently encouraging the sparsity in the Cholesky factor matrix \mathbf{L} and the estimated covariance matrix.

Denote by $\mathbf{x}^{(j)}$ the j th column of the data matrix $\mathbb{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)'$. Let $\mathbf{e}^{(j)}$ represent the residuals for the linear regression when $\mathbf{x}^{(j)}$ is treated as the response data, $j \geq 2$, and $\mathbf{e}^{(1)} = \mathbf{x}^{(1)}$. Let $\mathbb{W}^{(j)} = (\mathbf{e}^{(1)}, \dots, \mathbf{e}^{(j-1)})$ be the matrix containing the first $(j - 1)$ residuals. Now we construct the matrix \mathbf{L} by employing the Lasso penalty to select important predictor variables

$$\hat{\mathbf{l}}_j = \arg \min_{\mathbf{l}_j} \|\mathbf{x}^{(j)} - \mathbb{W}^{(j)} \mathbf{l}_j\|_2^2 + \lambda_j \|\mathbf{l}_j\|_1, \quad j = 2, \dots, p, \quad (43.11)$$

where $\lambda_j \geq 0$ is a tuning parameter and selected by cross-validation. $\mathbf{e}^{(j)} = \mathbf{x}^{(j)} - \mathbb{W}^{(j)} \hat{\mathbf{l}}_j$ is used to construct the residuals for the last column of $\mathbb{W}^{(j+1)}$. Then the element d_j^2 of the diagonal matrix $\mathbf{D} = \text{diag}(d_1^2, \dots, d_p^2)$ is estimated as the sample variance of $\mathbf{e}^{(j)}$

$$\hat{d}_j^2 = \widehat{\text{Var}}(\mathbf{e}^{(j)}) = \widehat{\text{Var}}(\mathbf{x}^{(j)} - \mathbb{W}^{(j)} \hat{\mathbf{l}}_j)$$

Consequently,

$$\hat{\boldsymbol{\Sigma}} = \hat{\mathbf{L}} \hat{\mathbf{D}} \hat{\mathbf{L}}' \quad (43.12)$$

is a sparse estimate for the covariance matrix $\boldsymbol{\Sigma}$. It is worth pointing out that one can also consider estimating $\mathbf{l}_2, \dots, \mathbf{l}_p$ in a joint fashion similar as in (43.7). But the estimation procedure will become more complicated since it involves latent variables.

Remark 2 For the optimization problem in (43.11) with $\lambda_j = 0$ and $n > p$, the estimated covariance matrix in (43.12) is equivalent to the sample covariance matrix.

43.2.3 MCD for Banded Matrix Estimation

In some applications such as assimilation [22] and social unrest study [35], the variables are strongly correlated with the ones that are close to them in the ordering, and the variables far apart in the ordering are weakly correlated. For example, in the random variable vector $\mathbf{X} = (X_1, \dots, X_p)'$, the variable X_3 may have a strong correlation with its neighbor variables X_1, X_2, X_4 , and X_5 , but it could be weakly correlated with variables X_j for $j > 5$. In this situation, the covariance or precision matrices are usually assumed to have a banded structure. The k th banded structure means that the first $k < p$ sub-diagonals elements of a matrix are non-zeroes with the rest elements being zeroes. Bickel and Levina [2] proposed a banded estimate for the covariance matrix by banding the sample covariance matrix. Note that their estimate cannot guarantee the positive definiteness. The MCD technique can also be used for estimating a banded matrix by banding the Cholesky matrix factor [28]. In this section, we focus on the estimation for banded precision matrix via the MCD. The Cholesky-based estimation for the banded covariance matrix follows the similar principle by considering the banded \mathbf{L} in (43.12).

The key idea is to band the Cholesky factor matrix \mathbf{T} . Recall that in the decomposition (43.1), the predictor variable X_j is regressed on all of its predecessors X_1, \dots, X_{j-1} . To accommodate a banded estimate of precision matrix, each predictor variable X_j is regressed only on its k previous variables X_{j-k}, \dots, X_{j-1} , for all $j = 2, \dots, p$. The index $j - k$ is interpreted to mean $\max(1, j - k)$. Hence, we solve the following $(p - 1)$ linear regression models instead of (43.1)

$$X_j = \sum_{i=j-k}^{j-1} a_{ji} X_i + \epsilon_j \quad (43.13)$$

to construct the Cholesky factors (\mathbf{T}, \mathbf{D}) . As a result, the unit lower triangular matrix \mathbf{T} , with ones as its diagonal and $(0, \dots, 0, a_{j,j-k}, \dots, a_{j,j-1})$ as its j th row, has a banded structure on its bottom left part. Then the estimate for the precision matrix $\boldsymbol{\Omega} = \mathbf{T}' \mathbf{D}^{-1} \mathbf{T}$ obtained from (43.13) has a k th banded structure. This approach is designated to estimate a banded matrix and guarantees the positive definiteness of the estimated matrix.

43.2.4 Adaptive Banding in the MCD

Note that the k th banded matrix estimation in Sect. 43.2.3 considers the banding width k to be the same for each vari-

able. In this section, we discuss a more flexible case where the band for each variable to be different. That is, we allow $k = k_j$ in the j th linear regression of (43.13), where the j th variable is regressed on its k_j closest predecessors with k_j depending on j . We call such a procedure as adaptive banding, which is useful when each variable may depend on an unknown number of its predecessors. It preserves sparsity in the resulting estimate of precision matrix and produces a better estimate by being able to adapt to the data. Next we introduce two types of adaptive banding. They use two different techniques to obtain an adaptively banded estimate for the Cholesky factor \mathbf{T} .

Adaptive Banding: Levina and Zhu [19] applied a nested Lasso penalty imposed on the negative normal log-likelihood function to produce matrix \mathbf{T} with adaptive banding k_j . Specially, the negative log-likelihood of the data, up to a constant, is

$$\begin{aligned} \ell(\boldsymbol{\Sigma}, \mathbf{x}_1, \dots, \mathbf{x}_n) &= n \log |\boldsymbol{\Sigma}| + \sum_{i=1}^n \mathbf{x}_i' \boldsymbol{\Sigma}^{-1} \mathbf{x}_i \\ &= n \log |\mathbf{D}| + \sum_{i=1}^n \mathbf{x}_i' \mathbf{T}' \mathbf{D}^{-1} \mathbf{T} \mathbf{x}_i \\ &= \sum_{j=1}^p \ell_j(d_j, \mathbf{a}_j, \mathbf{x}_1, \dots, \mathbf{x}_n), \end{aligned}$$

where \mathbf{a}_j stands for the vector of regression coefficients for the j th regression in the MCD and

$$\ell_j(d_j, \mathbf{a}_j, \mathbf{x}_1, \dots, \mathbf{x}_n) = n \log d_j^2 + \sum_{i=1}^n \frac{1}{d_j^2} \left(x_{ij} - \sum_{l=1}^{j-1} a_{jl} x_{il} \right)^2.$$

Minimizing the negative log-likelihood $\ell(\boldsymbol{\Sigma}, \mathbf{x}_1, \dots, \mathbf{x}_n)$ is equivalent to minimizing each $\ell_j(d_j, \mathbf{a}_j, \mathbf{x}_1, \dots, \mathbf{x}_n)$. Then one can consider to minimize

$$\ell_j(\boldsymbol{\Sigma}, \mathbf{x}_1, \dots, \mathbf{x}_n) + J(\mathbf{a}_j), \quad (43.14)$$

where $J(\mathbf{a}_j)$ is the nested Lasso penalty as

$$J(\mathbf{a}_j) = \lambda \left(|a_{j,j-1}| + \frac{|a_{j,j-2}|}{|a_{j,j-1}|} + \frac{|a_{j,j-3}|}{|a_{j,j-2}|} + \dots + \frac{|a_{j,1}|}{|a_{j,2}|} \right),$$

where $\lambda \geq 0$ is a tuning parameter, and we define $0/0 = 0$. The effect of variable selection by the nested Lasso penalty is that if the l th variable is not included in the j th regression ($a_{jl} = 0$), then all the subsequent variables ($l - 1$ through 1) are also excluded. Hence, the j th linear regression only has $k_j \leq j - 1$ closest predecessors, with values of k_j varying for each regression.

Since the expression in (43.14) is highly nonconvex and nonlinear, an iterative algorithm is developed to minimize it for constructing the Cholesky factor matrices \mathbf{T} and \mathbf{D} . Let $\hat{\mathbf{a}}_j^{(m)}$ and $\hat{d}_j^{(m)}$ represent the estimates of \mathbf{a}_j and d_j at the m th iteration. Then we repeat the following steps 1 and 2 until convergence.

Step 1: Given $\hat{\mathbf{a}}_j^{(m)}$, solve for $\hat{d}_j^{(m)}$

$$(\hat{d}_j^{(m)})^2 = \frac{1}{n} \sum_{i=1}^n \left(x_{ij} - \sum_{l=1}^{j-1} \hat{a}_{jl}^{(m)} x_{il} \right)^2.$$

Step 2: Given $\hat{\mathbf{a}}_j^{(m)}$ and $\hat{d}_j^{(m)}$, solve for $\hat{\mathbf{a}}_j^{(m+1)}$. Here we use the local quadratic approximation [9, 13]

$$|a_{jl}^{(m+1)}| \approx \frac{(a_{jl}^{(m+1)})^2}{2 |a_{jl}^{(m)}|} + \frac{|a_{jl}^{(m)}|}{2}.$$

Then the estimate of \mathbf{a}_j at iteration $m + 1$ is

$$\begin{aligned} \hat{\mathbf{a}}_j^{(m+1)} &= \arg \min_{\mathbf{a}_j} \frac{1}{(\hat{d}_j^{(m)})^2} \sum_{i=1}^n \left(x_{ij} - \sum_{l=1}^{j-1} a_{jl} x_{il} \right)^2 \\ &\quad + \frac{\lambda}{2} \left(\frac{a_{j,j-1}^2}{|\hat{a}_{j,j-1}^{(m)}|} + \sum_{l=1}^{j-2} \frac{a_{jl}^2}{|\hat{a}_{jl}^{(m)}| \cdot |\hat{a}_{j,l+1}^{(m)}|} \right). \end{aligned}$$

This is a quadratic optimization problem, which can be solved in closed form. Note that the algorithm needs an initial value for \mathbf{a}_j . One could use the least squares estimates when $p < n$. If $p > n$, we initialize with $\hat{a}_{jl}^{(0)} = \hat{a}_{jl}^*$, which are found by regressing X_j on X_l alone, for $l = 1, \dots, j - 1$.

Forward Adaptive Banding: Instead of imposing a nested Lasso penalty on the likelihood function, Leng and Li [18] proposed a forward adaptive banding approach to determine k_j for each linear regression by minimizing their modified Bayesian information criterion (BIC). Operationally, for the j th variable, we fit $j - 1$ nested linear models by regressing X_j on $X_{j-1}, \dots, X_{j-k_j}$ for $k_j = 1, \dots, j - 1$. Then the optimal k_j is chosen to minimize

$$\begin{aligned} \text{BIC} &= n \log |\hat{\boldsymbol{\Sigma}}| + \sum_{i=1}^n \mathbf{x}_i' \hat{\mathbf{T}}' \hat{\mathbf{D}}^{-1} \hat{\mathbf{T}} \mathbf{x}_i + C_n \log(n) \sum_{j=1}^p k_j \\ &= \sum_{j=1}^p \left\{ n \log \hat{d}_j^2 + \sum_{i=1}^n \frac{1}{\hat{d}_j^2} \left(x_{ij} - \sum_{l=1}^{k_j} a_{j,j-l} x_{i,j-l} \right)^2 + C_n \log(n) k_j \right\}, \end{aligned} \quad (43.15)$$

for all $k_j \leq \min\{n/(\log n)^2, j - 1\}$ ($j = 2, \dots, p$) with some diverging C_n . The expression in (43.15) implies that the banding width k_j can be chosen separately for each j . The penalty coefficient $C_n \log n$ is set to different values to accommodate the diverging number of predictor variables p . Leng and Li [18] proved that this modified BIC is model selection consistent under some regular conditions. The advantage of this approach is that we determine the value of k_j by only fitting a sequence of linear models. Therefore, it can be easily and efficiently implemented.

43.3 Ordering Issue in the MCD

Although the MCD approach has been widely investigated for the matrix estimation, only a few work contributed to solve a potential problem, the ordering issue. From the decompositions (43.1) and (43.10), it is clear to see that different orderings of the predictor variables X_1, \dots, X_p used in the MCD would lead to different linear regressions. If a regularization is adopted to shrink the coefficients of the linear regressions, such as the Lasso penalty in (43.6), the Cholesky factor matrices estimates would be different under different variable orderings, thus leading to the different estimates for the covariance and precision matrices. As a statistical decision problem, ordering variables is quite challenging. In practice, the MCD method can be suitably used to estimate a matrix without this ordering issue when the predictor variables have a natural ordering among themselves, as in time series, longitudinal data, or spatial data. However, there are a large number of applications where such a natural ordering is not available or the variable ordering cannot be determined before the analysis, for example, gene expressions, financial, and economic data. In these cases, one may need to first determine a proper ordering among the variables before employing the MCD method. Next, we introduce three commonly used techniques to tackle this issue. In this part, we focus on the ordering issue in the MCD for the estimation of precision matrix. Ordering issue for Cholesky-based estimation of covariance matrix can be addressed in a similar fashion.

BIC: A search algorithm for ordering the predictor variables based on the BIC criterion can be easily implemented. In each step, a new variable is selected into the ordering with the smallest value of BIC when regressing it on the rest of the candidate variables. Specifically, suppose we want to construct an ordering for p variables X_1, \dots, X_p . In the first step, each variable $X_j, j = 1, \dots, p$ is regressed on the rest of variables, producing p values of BIC from p linear regressions. Then the first variable determined into the ordering is the response variable of the linear regression that gives the minimum value of BIC. This selected variable is denoted by X_{i_p} , and it is assigned to the p th position of the

ordering. All the variables excluding the selected variable X_{i_p} consist of the candidate set \mathcal{C} . In the second step, from the set \mathcal{C} , each variable is regressed on the rest of variables, producing $p - 1$ values of BIC from $p - 1$ linear regressions. Then the second variable, denoted by $X_{i_{p-1}}$, determined into the ordering is the response variable of the linear regression that gives the minimum value of BIC. The variable $X_{i_{p-1}}$ is assigned to the $(p - 1)$ th position of the ordering. Then all the variables excluding X_{i_p} and $X_{i_{p-1}}$ compose of the candidate set for the next round.

To sum up this procedure, let $\mathcal{C} = \{X_{i_1}, \dots, X_{i_k}\}$ be the candidate set of variables, and there are $p - k$ variables already chosen into the ordering. By regressing each $X_j, j = i_1, \dots, i_k$ on the rest of the variables in \mathcal{C} , the variable corresponding to the minimum BIC value among the k regressions is assigned to the k th position of the ordering. Then the ordering created by this procedure is used for the MCD.

BPA: A popular method to recover the variable ordering for the autoregressive model is the best permutation algorithm (BPA) developed by Rajaratnam and Salzman [25]. It is formulated as a well-defined optimization problem where the optimal ordering is determined as the one minimizing the sum of squared diagonal entries of the Cholesky factor matrix D in the MCD. For the convenience of presentation, define a permutation mapping $\pi : \{1, \dots, p\} \rightarrow \{1, \dots, p\}$ by

$$(\pi(1), \pi(2), \dots, \pi(p)). \tag{43.16}$$

Denote the corresponding permutation matrix by P_π of which the entries in the j th column are all 0 except taking 1 at position $\pi(j)$. Let S_p be the symmetric group of all permutations of the integers $1, \dots, p$. For a given $\pi \in S_p$, let Ω_π be the precision matrix corresponding to the variable ordering π and

$$\Omega_\pi = T'_\pi D_\pi^{-1} T_\pi$$

be its MCD. Since the conditional variances can be used as natural measures to quantify the extent to which the variability of a random variable is explained by the ones that precede it. Hence, the conditional variances give a sense of closeness between variables. From this viewpoint, the BPA is to find an ordering π^* in S_p to minimize $\|D_{\pi^*}\|_F^2$, where $\|\cdot\|_F$ represents the Frobenius norm. Then $\hat{\Omega} = P_{\pi^*} \hat{\Omega}_{\pi^*} P_{\pi^*}' = P_{\pi^*} \hat{T}'_{\pi^*} \hat{D}_{\pi^*}^{-1} \hat{T}_{\pi^*} P_{\pi^*}'$ is an estimate of precision matrix based on BPA. Rajaratnam and Salzman [25] showed the consistency of this approach in recovering the natural order of variables in underlying autoregressive models.

Ordering-averaged method (OAM): To address the problem that the ordering of variables is often not available in practice, we consider a Cholesky-based model averaging idea [40] by averaging a representative set of individual matrix estimates obtained from random permutations of the variable orderings. This method does not require any prior knowledge

of the orderings of variables; hence, it is suitable for the case where there is no natural ordering among the variables, or such an ordering cannot be easily determined. Use the definition of permutation mapping π in (43.16), a precision matrix estimate under π is $\mathbf{\Omega}_\pi = \mathbf{T}'_\pi \mathbf{D}_\pi^{-1} \mathbf{T}_\pi$. Transforming back to the original ordering, we can estimate $\mathbf{\Omega}$ as

$$\hat{\mathbf{\Omega}} = \mathbf{P}_\pi \hat{\mathbf{\Omega}}_\pi \mathbf{P}'_\pi = \mathbf{P}_\pi \mathbf{T}'_\pi \mathbf{D}_\pi^{-1} \mathbf{T}_\pi \mathbf{P}'_\pi. \quad (43.17)$$

By incorporating several permutations of π s, one can obtain a pool of precision matrix estimates. Taking the average of these estimates leads to an ordering-averaged estimation. In practice, a modest number of permutations are sufficient to serve this purpose. Therefore, we randomly generate multiple permutations $\pi_k, k = 1, \dots, M$ and obtain the corresponding estimates $\hat{\mathbf{\Omega}}$ in (43.17), denoted by $\hat{\mathbf{\Omega}}_k$ for the permutation π_k . The model averaging estimate of $\mathbf{\Omega}$ is

$$\hat{\mathbf{\Omega}}_{OAM} = \frac{1}{M} \sum_{k=1}^M \hat{\mathbf{\Omega}}_k. \quad (43.18)$$

Similarly, it is very convenient to apply the OAM for the estimation of the covariance matrix. Combining Eq. (43.9) and the averaging idea of (43.18), we have the model averaging estimate of $\mathbf{\Sigma}$ as

$$\begin{aligned} \hat{\mathbf{\Sigma}}_{OAM} &= \frac{1}{M} \sum_{k=1}^M \hat{\mathbf{\Sigma}}_k = \frac{1}{M} \sum_{k=1}^M \mathbf{P}_k \hat{\mathbf{\Sigma}}_k \mathbf{P}'_k \\ &= \frac{1}{M} \sum_{k=1}^M \mathbf{P}_k \hat{\mathbf{T}}_k^{-1} \hat{\mathbf{D}}_k (\hat{\mathbf{T}}_k')^{-1} \mathbf{P}'_k, \end{aligned}$$

where $\hat{\mathbf{T}}_k, \hat{\mathbf{D}}_k$, and $\hat{\mathbf{\Sigma}}_k$ represent the estimates of \mathbf{T}, \mathbf{D} , and $\mathbf{\Sigma}$ under the permutation π_k . According to the finite population sampling survey theory [7], the selection of permutations π_k is not essential when we use a reasonable size M . Although choosing a larger M would further reduce the variability of the OAM estimate, Zheng et al. [40] showed that a modest number $M = 30$ is seen to lead to stable results.

43.4 Real Applications

The covariance and precision matrices have been widely used in various areas such as portfolio selection, risk assessment, principle component analysis, social network, graphical models, classification, and so forth. In this section, real-data examples are used to illustrate the application of the MCD approach for the estimation of covariance and precision matrices.

43.4.1 MCD for Varying Covariance Matrix Estimation in Multivariate Time Series

In the financial management with multivariate time series, a major task is to estimate the time-varying covariance matrices $\{\mathbf{\Sigma}_t\}$ based on the (conditionally) independently distributed data $\mathbf{x}_t \sim N(\mathbf{0}, \mathbf{\Sigma}_t), t = 1, 2, \dots, n$. The data \mathbf{x}_t can be viewed as the returns of p assets in a portfolio at time t .

By the decomposition (43.9), the estimation of time-varying covariance matrix is given by

$$\mathbf{\Sigma}_t = \mathbf{T}^{-1} \mathbf{D}_t \mathbf{T}'^{-1},$$

where the Cholesky factor matrix $\mathbf{T} = \mathbf{T}_t$ for all $t = 1, 2, \dots, n$ is assumed to be time-invariant to reduce a large number of parameters. For each element of the diagonal matrix $\mathbf{D}_t = \text{diag}(d_{1,t}^2, \dots, d_{p,t}^2)$, the $\log d_{j,t}^2, j = 1, 2, \dots, p$, is modeled using the log-GARCH(u, v) defined recursively in time as

$$\begin{aligned} \log d_{j,t}^2 &= \beta_0^{(j)} + \sum_{i=1}^v \left(\alpha_{i+}^{(j)} 1_{\{\epsilon_{j,t-i} > 0\}} + \alpha_{i-}^{(j)} 1_{\{\epsilon_{j,t-i} < 0\}} \right) \log \epsilon_{j,t-i}^2 \\ &\quad + \sum_{k=1}^u \beta_k^{(j)} \log d_{j,t-k}^2. \end{aligned} \quad (43.19)$$

where $1_{\{\cdot\}}$ is the indicator function and $\beta_0^{(j)}, \beta_k^{(j)}, \alpha_{i+}^{(j)}, \alpha_{i-}^{(j)}$ are corresponding coefficients. The quasi-maximum likelihood approach in Francq and Zakoian [11] is used to fit the model (43.19). Therefore, we combine the MCD method and the log-GARCH model to analyze a stock return data from the Standard and Poor's 100 index (S&P100). The data set comprises of $n = 436$ returns and $p = 97$ stocks weekly recorded from August 23, 2004, to December 12, 2012. For simplicity, the log-GARCH (1, 1) model is used to estimate \mathbf{D}_t .

To measure the accuracy of the covariance matrix estimates, we consider the following loss functions: the entropy loss Δ_{1t} , the Kullback-Leibler loss Δ_{2t} , and the quadratic loss functions Δ_{3t} (up to some scale) defined as

$$\begin{aligned} \Delta_{1t} &= \frac{1}{p} \left[\text{tr}[\mathbf{\Sigma}_t^{-1} \hat{\mathbf{\Sigma}}_t] - \log |\mathbf{\Sigma}_t^{-1} \hat{\mathbf{\Sigma}}_t| - p \right], \\ \Delta_{2t} &= \frac{1}{p} \left[\text{tr}[\hat{\mathbf{\Sigma}}_t^{-1} \mathbf{\Sigma}_t] - \log |\hat{\mathbf{\Sigma}}_t^{-1} \mathbf{\Sigma}_t| - p \right], \\ \Delta_{3t} &= \frac{1}{p} \left[\text{tr}(\hat{\mathbf{\Sigma}}_t^{-1} \mathbf{\Sigma}_t - \mathbf{I})^2 \right]. \end{aligned}$$

We also use the mean absolute error and mean squared error loss functions given by

$$\text{MAE}_t = \frac{1}{p^2} \sum_{i=1}^p \sum_{j=1}^p |\hat{\omega}_{ij;t} - \omega_{ij;t}| \quad \text{and}$$

$$\text{MSE}_t = \frac{1}{p^2} \sum_{i=1}^p \sum_{j=1}^p (\hat{\omega}_{ij;t} - \omega_{ij;t})^2,$$

where $\hat{\Sigma}_t = (\hat{\omega}_{ij;t})_{p \times p}$ stands for the estimate of the covariance matrix $\Sigma_t = (\omega_{ij;t})_{p \times p}$, $t = 1, \dots, n$. For each loss measure, we report their averages over the time t as $\text{MAE} = \sum_{t=1}^n \text{MAE}_t/n$, $\text{MSE} = \sum_{t=1}^n \text{MSE}_t/n$, and $\Delta_i = \sum_{t=1}^n \Delta_{it}/n$, $i = 1, 2, 3$.

Since the true realized covariance matrix Σ_t is unknown, we employ a moving blocks approach to get a reliable proxy [23]. Table 43.1 reports the averaged loss measures over time t and their standard errors in parenthesis for methods including ORIG, BIC, BPA, DCC, and OAM. ORIG represents the Cholesky-based method for the estimation of covariance matrix based on the original ordering of variables. Similarly, BIC and BPA are the Cholesky-based methods for the estimation of covariance matrix based on the BIC and BPA to determine the ordering of variables, respectively. OAM stands for the Cholesky-based estimate of covariance matrix using the ordering-averaged model. DCC represents for the dynamic conditional correlation GARCH model, which is a popular tool to fit time series data in finance. It imposes a dynamic structure on the conditional correlation matrices.

It is clear to see from Table 43.1 that the OAM estimate based on the MCD gives the best performance regarding all the loss functions. The possible reason is that the stocks have no natural ordering among themselves. For example, it is not reasonable to order the stocks of Apple Inc., Dow Chemical Co., Microsoft Corp, and Bank of America Corp. Hence, the ordering-averaged model shows a relatively accurate estimation. The second best is the BPA, which is slightly inferior to the OAM. The DCC model does not provide accurate estimates for the time-varying covariance matrices compared with the OAM and BPA for this data set. The ORIG and BIC methods produce large losses, especially in terms of the quadratic loss Δ_3 and MSE. Overall, the MCD approach is suitable for the time-varying covariance matrix estimation for this time-series data.

43.4.2 MCD for Portfolio Optimization

Now we consider a portfolio optimization process that determines the portfolio allocation of multiple assets to minimize the portfolio variance. The risk of a portfolio $w = (w_1, \dots, w_p)$ is measured by the variance $w' \Sigma w$ of its return, where $w_i \geq 0$ and $\sum_{i=1}^p w_i = 1$. The estimated minimum variance portfolio optimization problem is formulated as

$$\min_w w' \hat{\Sigma} w \tag{43.20}$$

$$\text{s.t.} \quad \sum_{i=1}^p w_i = 1,$$

where $\hat{\Sigma}$ is an estimate of the true covariance matrix Σ of asset returns.

We illustrate the performances of each method, including ORIG, BIC, BPA, OAM, and DCC, applied into the portfolio optimization problem using the same data set of 97 stock returns from S&P100 as in Sect. 43.4.1. The first t observations are used to estimate the Cholesky factor matrices (T, D_t) and then predict the covariance matrices $\hat{\Sigma}_{t+1} = \hat{T}^{-1} \hat{D}_{t+1} \hat{T}'^{-1}$ at time $t + 1$, $t = 350, 351, \dots, 435$. Note that the value of $d_{j,t+1}^2$, which is the diagonal element of D_{t+1} , can be estimated from Eq. (43.19) by generating $\hat{\epsilon}_{j;t} = \hat{d}_{j;t} \eta_t$, where η_t is a random variable with mean 0. For this application, we choose $\eta_t \sim t_{df=5}$ distribution. The estimated portfolio \hat{w}_{t+1} is the solution of (43.20) by replacing $\hat{\Sigma}$ with $\hat{\Sigma}_{t+1}$. In practice, the researchers care not only the portfolio risk in (43.20) but also the reward and the information ratio (reward to risk). Hence, the performance measures of interest are the average annual realized return

$$\text{AVG} = \frac{1}{86} \sum_{t=350}^{435} 52 * \hat{w}'_{t+1} x_{t+1},$$

their standard deviation (SD), and the information ratio AVG/SD . Notice that the optimization (43.20) is designed to minimize the portfolio variance rather than to maximize the expected return or the information ratio. Therefore, any portfolio should be primarily evaluated by how successfully it achieves the minimum SD. A large realized return and a high value of information ratio are naturally also desirable

Table 43.1 The averages and standard errors (in parenthesis) of loss measures for the weekly returns of 97 stocks

	Δ_1	Δ_2	Δ_3	MAE	MSE
ORIG	3.397 (0.054)	3.196 (0.036)	1017 (28.35)	6.380 (0.089)	90.61 (2.167)
BIC	3.017 (0.077)	3.376 (0.049)	1176 (32.69)	9.004 (0.486)	364.0 (87.92)
BPA	0.747 (0.029)	0.446 (0.010)	14.94 (0.970)	3.877 (0.096)	36.12 (1.594)
DCC	0.902 (0.049)	0.825 (0.029)	142.3 (11.35)	5.338 (0.223)	172.8 (27.32)
OAM	0.681 (0.024)	0.335 (0.005)	6.183 (0.344)	3.687 (0.102)	35.27 (1.897)

Table 43.2 The comparison of portfolio performances for the weekly returns of 97 stocks

	ORIG	BIC	BPA	DCC	OAM
AVG	8.078	10.93	10.70	8.738	9.840
SD	14.59	17.09	8.007	8.800	7.230
AVG/SD	0.554	0.640	1.336	0.993	1.361

but should be considered of secondary importance from the point of view of evaluating the quality of a covariance matrix estimate.

Table 43.2 summarizes the portfolio performances for each method in terms of AVG, SD, and the information ratio. The OAM estimate provides the smallest SD and the largest information ratio, followed by the BPA method. Although the BIC produces the highest value of the realized return, it has the worst portfolio risk among all the approaches, hence resulting in a low information ratio. The ORIG gives inferior performance with relatively small AVG and large SD. The DCC model does not perform well as the OAM and BPA, but it is better than the BIC and ORIG for this set of data.

43.4.3 MCD for Linear Discriminant Analysis

Linear discriminant analysis (LDA) is a commonly used classification method in statistics and machine learning to construct a decision boundary via a linear combination of predictor variables that separates two or more classes of objects. For a multiple-class discriminant problem, each observation \mathbf{x} belongs to some class $k \in 1, 2, \dots, K$. Let $Y \in \{1, 2, \dots, K\}$ represent K classes. Under the assumption that the conditional density function $f(\mathbf{x}|Y = k)$ follows a normal distribution $N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma})$, the LDA classification rule is

$$\eta_k(\mathbf{x}) = \mathbf{x}'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k - \frac{1}{2}\boldsymbol{\mu}_k'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k + \log \pi_k, \quad (43.21)$$

where π_k is the prior probability for class k . The observation \mathbf{x} is assigned to the class k^* if $k^* = \underset{k}{\arg \max} \eta_k(\mathbf{x})$.

There are several unknown parameters in the classification rule (43.21), the population mean $\boldsymbol{\mu}_k$, precision matrix $\boldsymbol{\Sigma}^{-1}$, and prior probability π_k . We estimate them from the training data set. Let C_k be the set composed of the training observations belonging to the class k . Denote by $\hat{\boldsymbol{\mu}}_k$ the $p \times 1$ vector of the sample mean for the training data in class k . Let $\hat{\boldsymbol{\Sigma}}_{LDA} = \frac{1}{n-K} \sum_{k=1}^K \sum_{i \in C_k} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_k)(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_k)'$ be the estimated within-class covariance matrix based on the training data. Then the estimated LDA classification rule is

$$\mathbf{x}'\hat{\boldsymbol{\Sigma}}_{LDA}^{-1}\hat{\boldsymbol{\mu}}_k - \frac{1}{2}\hat{\boldsymbol{\mu}}_k'\hat{\boldsymbol{\Sigma}}_{LDA}^{-1}\hat{\boldsymbol{\mu}}_k + \log \hat{\pi}_k,$$

Table 43.3 Misclassification rates (in percentage) under 50 times randomly splitting for hand movement data

Method	BIC	BPA	Glasso	GLDA	DLDA	OAM
Misclassification	40.1	39.0	39.1	51.0	46.3	38.9
SE	0.5	0.5	0.6	0.5	0.6	0.5

where $\hat{\pi}_k$ is the frequency of class k in the training data set. It is clear that the estimation accuracy of the covariance or precision matrix will have a profound effect on the accuracy of the classification accuracy of the LDA methods. See the derivation in the appendix for the details. For the high-dimensional data, the $\hat{\boldsymbol{\Sigma}}_{LDA}$ is often unstable or singular. Hence some other estimates of $\boldsymbol{\Sigma}^{-1}$ are used such as the generalized inverse of the within-class covariance matrix (GLDA) or Glasso estimate [38]. In this section, we consider $\boldsymbol{\Sigma}^{-1}$ estimated by some Cholesky-based precision matrix estimation methods such as BIC, OAM, and BPA and examine their performances under the LDA framework. We also present the classification results obtained by estimating $\boldsymbol{\Sigma}^{-1}$ from Glasso, GLDA, and DLDA methods. The DLDA assumes the off-diagonal elements of the precision matrix to be zeroes.

We apply each method into the hand movement data set [29], which contains 15 classes with 24 observations in each class. Every class refers to a hand movement type. The hand movement is represented as a two-dimensional curve performed by the hand in a period of time, where each curve is mapped in a representation with 90 predictor variables. The whole data set is randomly split into a training set with 160 observations and testing set with the rest 200 observations. The 160 observations in the training set are used to estimate the population means $\boldsymbol{\mu}_k$, the population covariance matrix $\boldsymbol{\Sigma}$, and the prior probability π_k in the classification rule (43.21). Then the testing data are used to compute the misclassification rate for each method. The above procedure is repeated 50 times, and Table 43.3 reports the averages and standard errors (SE) of the misclassification rate in percentage for each method.

We see that the Cholesky-based estimates of BIC, OAM, BPA, and Glasso estimate perform comparably with respect to the misclassification rate. The DLDA produces an inferior performance result, possibly due to the reason that the underlying precision matrix of the 90 predictor variables might not be a diagonal structure. In other words, some variables are not conditional independent with each other. The GLDA does not perform well for this data set. It only has a half chance of assigning a new object to the correct class. Note that the misclassification rate of each method is relatively large, since the data contain 15 classes, which makes the discriminant analysis more difficult.

43.5 Numerical Study

In this section, we provide several numerical examples to further illustrate the Cholesky-based estimation of covariance and precision matrices. Here we consider the following two covariance and two precision matrix structures.

Model 1. $\Sigma_1 = \text{MA}(0.5, 0.3)$. The diagonal elements are 1 with the first sub-diagonal elements 0.5 and the second sub-diagonal elements 0.3.

Model 2. Σ_2 is generated by randomly permuting rows and corresponding columns of Σ_1 .

Model 3. Ω_2 is a diagonal matrix with its diagonal elements the inverse of vector $(p, p - 1, p - 2, \dots, 1)'$.

Model 4. $\Omega_1 = \text{AR}(0.5)$. The conditional covariance between any two random variables X_i and X_j is fixed to be $0.5^{|i-j|}$, $1 \leq i, j \leq p$.

Model 1 is a banded sparse matrix, while **Model 2** is an unstructured sparse matrix. **Model 3** is a diagonal matrix. **Model 4** is an autoregressive structure that has homogeneous variances and correlations declining with distance. This model is more dense than the other models. For each model, data x_1, \dots, x_n are generated independently from the normal distribution $N(\mathbf{0}, \Sigma)$ with $n = 50$ and $p \in \{30, 50\}$. We use the same loss functions as in the application section to evaluate the accuracy of the estimates. Besides, to examine the performances of the estimates in catching the sparse structure, the false selection loss (FSL) is used, which is the summation of false positive (FP) and false negative (FN). The FSL is computed in percentage as $(\text{FP} + \text{FN}) / p^2$. Tables 43.4 and 43.5 report the averages of loss measures and their

corresponding standard errors (in parenthesis), respectively, for each method based on 50 replications. The dashed lines in the tables represent the corresponding values not available due to matrix singularity.

In Table 43.4, we compare the performances of the sample covariance matrix S , BIC, BPA, OAM, and RLZ. The RLZ represents the covariance estimator proposed by Rothman et al. [28] (see details in Sect. 43.2.3), which is designated for the banded matrix estimation. From the table, it is seen that the ordering-averaged method OAM gives better results than the sample covariance matrix, BIC and BPA with respect to $\Delta_1, \Delta_2, \Delta_3$, MAE, and MSE, since it takes advantage of multiple variable orderings over one single ordering. However, the OAM does not capture the sparse structure, which is destroyed by the average operation in (43.18). The BIC and BPA can have some sparsity regarding FSL due to the Lasso regularization when constructing the Cholesky factors. But their FSL are worse than that of RLZ, which directly forces most of elements in the Cholesky factors to be zeroes. In addition, it is not surprising to observe that the RLZ performs better for the covariance matrix Σ_1 than Σ_2 , since Σ_2 is no longer a banded matrix after random permutations of rows and columns.

Table 43.5 presents the results obtained by the BIC, BPA, OAM, and Glasso methods. The ordering-averaged method OAM performs generally well but produces no sparsity for the matrix. In contrast, the Glasso estimate is good at catching the sparse structure as seen from FSL. The Glasso method [38] imposes an L_1 type penalty on the negative likelihood function of Ω , hence encouraging the sparsity in the esti-

Table 43.4 The averages and standard errors of estimates for covariance matrix Σ

			Δ_1	Δ_2	Δ_3	MAE	MSE	FSL(%)
Σ_1	$p = 30$	S	12.3 (0.12)	38.7 (1.11)	89.9 (4.54)	3.51 (0.04)	19.9 (0.46)	84.0 (0.02)
		BIC	7.11 (0.12)	11.0 (0.53)	8.11 (0.84)	1.74 (0.02)	10.6 (0.19)	52.0 (0.80)
		BPA	5.94 (0.10)	7.97 (0.28)	4.46 (0.35)	1.53 (0.02)	8.78 (0.19)	45.9 (0.91)
		OAM	4.60 (0.07)	5.24 (0.11)	1.58 (0.11)	1.56 (0.02)	7.88 (0.17)	83.6 (0.04)
		RLZ	9.34 (0.09)	6.58 (0.06)	0.35 (0.03)	1.04 (0.01)	8.51 (0.08)	6.22 (0.01)
	$p = 50$	S	–	–	–	5.77 (0.04)	53.0 (0.70)	90.2 (0.01)
		BIC	15.7 (0.30)	52.8 (8.35)	147 (49.0)	1.96 (0.02)	20.3 (0.24)	42.6 (0.70)
		BPA	13.0 (0.26)	20.3 (1.36)	16.2 (2.62)	1.82 (0.03)	18.4 (0.33)	41.1 (0.93)
		OAM	9.31 (0.10)	10.0 (0.16)	2.88 (0.19)	1.79 (0.02)	15.6 (0.25)	89.2 (0.06)
		RLZ	16.2 (0.10)	11.4 (0.08)	0.65 (0.07)	1.06 (0.01)	14.3 (0.13)	3.84 (0.01)
Σ_2	$p = 30$	S	12.4 (0.12)	38.6 (1.11)	88.2 (4.56)	3.53 (0.03)	19.8 (0.37)	84.0 (0.01)
		BIC	7.31 (0.14)	11.3 (0.51)	8.49 (0.84)	1.75 (0.02)	10.3 (0.17)	52.3 (0.66)
		BPA	5.81 (0.12)	7.58 (0.31)	4.06 (0.43)	1.49 (0.02)	8.40 (0.18)	46.5 (1.02)
		OAM	4.61 (0.08)	5.14 (0.11)	1.43 (0.12)	1.54 (0.01)	7.50 (0.16)	83.5 (0.05)
		RLZ	17.0 (0.17)	10.2 (0.05)	0.28 (0.04)	1.68 (0.01)	18.9 (0.07)	15.6 (0.01)
	$p = 50$	S	–	–	–	5.73 (0.04)	52.7 (0.74)	90.2 (0.01)
		BIC	16.0 (0.32)	50.1 (7.06)	121 (33.6)	1.97 (0.02)	20.4 (0.24)	43.2 (0.75)
		BPA	12.8 (0.23)	20.3 (1.51)	16.6 (3.15)	1.83 (0.03)	18.5 (0.29)	40.9 (0.96)
		OAM	9.39 (0.09)	10.3 (0.15)	3.08 (0.19)	1.81 (0.01)	15.9 (0.23)	89.2 (0.05)
		RLZ	31.6 (0.25)	19.8 (0.08)	0.75 (0.09)	1.90 (0.01)	36.5 (0.08)	11.4 (0.01)

Table 43.5 The averages and standard errors of estimates for precision matrix Ω

			Δ_1	Δ_2	Δ_3	MAE	MSE	FSL(%)
Ω_1	$p = 30$	BIC	2.79 (0.26)	1.84 (0.09)	1.34 (0.23)	0.10 (0.01)	0.45 (0.09)	27.4 (2.09)
		BPA	2.45 (0.13)	1.72 (0.06)	0.99 (0.12)	0.13 (0.01)	0.42 (0.05)	21.7 (1.69)
		OAM	1.99 (0.14)	1.40 (0.06)	1.19 (0.16)	0.10 (0.01)	0.27 (0.04)	79.8 (0.84)
		Glasso	2.86 (0.06)	5.48 (0.16)	1.65 (0.08)	0.07 (0.01)	0.80 (0.01)	8.24 (0.34)
	$p = 50$	BIC	14.0 (3.78)	3.97 (0.22)	22.6 (12.8)	0.26 (0.09)	4.64 (2.57)	31.0 (2.41)
		BPA	5.32 (0.47)	3.22 (0.15)	2.62 (0.39)	0.13 (0.02)	0.79 (0.34)	19.3 (1.21)
		OAM	4.36 (0.28)	2.73 (0.09)	3.11 (0.26)	0.10 (0.01)	0.27 (0.03)	73.2 (1.28)
		Glasso	5.45 (0.07)	11.9 (0.23)	2.74 (0.12)	0.05 (0.01)	1.00 (0.01)	7.97 (0.34)
Ω_2	$p = 30$	BIC	9.03 (0.43)	6.05 (0.11)	5.92 (0.61)	3.34 (0.20)	42.3 (5.52)	45.7 (0.31)
		BPA	6.29 (0.19)	5.20 (0.09)	2.84 (0.23)	2.29 (0.06)	17.5 (0.87)	44.8 (0.48)
		OAM	5.51 (0.17)	3.98 (0.06)	3.68 (0.27)	2.15 (0.05)	12.9 (0.63)	46.7 (0.04)
		Glasso	6.08 (0.10)	14.0 (0.37)	1.71 (0.07)	2.28 (0.01)	24.0 (0.24)	45.2 (0.17)
	$p = 50$	BIC	32.6 (5.21)	12.8 (0.26)	61.8 (30.7)	8.09 (1.93)	221 (29.9)	55.3 (0.59)
		BPA	14.2 (0.87)	10.8 (0.16)	8.54 (1.46)	3.21 (0.39)	81.1 (42.4)	44.0 (0.55)
		OAM	14.1 (0.59)	8.63 (0.12)	13.0 (1.06)	3.47 (0.21)	49.7 (7.84)	65.4 (0.03)
		Glasso	11.7 (0.12)	29.0 (0.47)	3.52 (0.09)	2.40 (0.01)	44.1 (0.25)	30.6 (0.07)

mated precision matrix. Because the underlying precision matrix Ω_2 is denser than Ω_1 , the Glasso method appears to perform better with respect to FSL for Ω_1 . Additionally, the Glasso also performs well regarding the loss measures Δ_3 and MAE. We also note that the BPA gives a superior performance than BIC estimate. This superiority is more evident for the precision matrix Ω_2 compared with Ω_1 , since Ω_2 is an autoregressive model.

43.6 Discussion

This chapter reviews the modified Cholesky decomposition (MCD) method for the estimation of covariance and precision matrices. It is seen that the MCD method has the flexibility of handling matrix estimation by transforming the problem into a sequence of regression-based problems. The sparsity can be easily imposed on the Cholesky factor matrices via the linear regressions and hence encouraging the sparse structure in the matrix estimate. The Cholesky-based methods guarantee the positive definiteness of the estimated matrix. Note that such Cholesky-based methods often require the knowledge on the ordering of variables when estimating a sequence of regressions. To address this issue, we thoroughly discuss the ordering issue in the MCD approach and examine several solutions to this ordering issue from the literature.

It is worth pointing out that the use of Cholesky-based approach is not restricted for estimating covariance and precision matrices under the conventional setting. The idea of MCD can also be used in constructing the covariance function in spatial analysis such as Kriging and Gaussian process [8], especially when involving the qualitative input variables. Because of the Cholesky decomposition on the matrix, the induced covariance matrix would have attractive

property in the Gaussian process modeling. Another direction of using the MCD is the multi-response regression, where the multivariate responses have certain dependency structures. Under this situation, the MCD method of estimating the covariance or precision matrix for the multivariate responses will be coupled with the estimation of regression coefficients.

Acknowledgments The authors would like to thank the editor and reviewers for the constructive and insightful comments, which have significantly enhanced the quality of this article.

Appendix

Proof of Remark 1 and 2 Since the conclusions of Remarks 1 and 2 are much similar, we only provide the proof of Remark 2 here. Assume that there are n independent and identically distributed observations $\mathbf{x}_1, \dots, \mathbf{x}_n$, which are centered. Let $\mathbf{S} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i'$ be the sample covariance matrix and assume that \mathbf{S} is non-singular since $n > p$. We denote $\hat{\Sigma}_0$ as the estimated covariance matrix from (43.12) with tuning parameters equal to zeroes in (43.11). Then Remark 2 states that $\hat{\Sigma}_0 = \mathbf{S}$ in spite of any permutation of $\mathbf{x}_1, \dots, \mathbf{x}_n$. Below is the proof.

Based on the sequential regression of (43.11), it is known that the first step is $X_1 = \epsilon_1$. It means that

$$e_{i1} = x_{i1}, 1 \leq i \leq n, \text{ and } \hat{\sigma}_1^2 = \frac{1}{n} \sum_{i=1}^n e_{i1}^2$$

Then the second step is to consider $X_2 = l_{21}\epsilon_1 + \epsilon_2$, which provides

$$\hat{l}_{21} = \frac{\sum_{i=1}^n x_{i2}e_{i1}}{\sum_{i=1}^n e_{i1}^2}, \quad e_{i2} = x_{i2} - \hat{l}_{21}e_{i1}, \quad 1 \leq i \leq n$$

$$\hat{\sigma}_2^2 = \frac{1}{n} \sum_{i=1}^n e_{i2}^2, \quad \sum_{i=1}^n e_{i2}e_{i1} = 0$$

In general, the j th step is to consider the regression problem as

$$X_j = \sum_{k < j} l_{jk} \epsilon_k + \epsilon_j$$

and we can obtain

$$\hat{l}_{j1} = \frac{\sum_{i=1}^n x_{ij}e_{i1}}{\sum_{i=1}^n e_{i1}^2}, \dots, \hat{l}_{jk} = \frac{\sum_{i=1}^n x_{ij}e_{ik}}{\sum_{i=1}^n e_{ik}^2}, \dots, \hat{l}_{j,j-1} = \frac{\sum_{i=1}^n x_{ij}e_{i,j-1}}{\sum_{i=1}^n e_{i,j-1}^2}$$

$$e_{ij} = x_{ij} - \sum_{k < j} \hat{l}_{jk} e_{ik}, \quad 1 \leq i \leq n$$

$$\hat{\sigma}_j^2 = \frac{1}{n} \sum_{i=1}^n e_{ij}^2, \quad \sum_{i=1}^n e_{ij}e_{i1} = 0, \dots, \sum_{i=1}^n e_{ij}e_{i,j-1} = 0$$

Therefore, we can express the (s, t) entry of the covariance matrix estimate using the regression coefficients as

$$(\hat{\Sigma})_{st} = (\hat{\mathbf{L}}\hat{\mathbf{D}}\hat{\mathbf{L}}^T)_{st} = \sum_{u=1}^{\min(s,t)} \hat{l}_{su}\hat{l}_{tu}\hat{\sigma}_u^2 \quad (\hat{l}_{uu} = 1).$$

Note that

$$x_{is} = \sum_{u=1}^s \hat{l}_{su}e_{iu} \quad (\hat{l}_{uu} = 1), \quad 1 \leq i \leq n,$$

$$x_{it} = \sum_{v=1}^t \hat{l}_{tv}e_{iv} \quad (\hat{l}_{vv} = 1), \quad 1 \leq i \leq n,$$

and the (s, t) entry of the sample covariance matrix is

$$(\mathbf{S})_{st} = \frac{1}{n} \sum_{i=1}^n x_{is}x_{it} = \frac{1}{n} \sum_{i=1}^n \left(\sum_{u=1}^s \hat{l}_{su}e_{iu} \right) \left(\sum_{v=1}^t \hat{l}_{tv}e_{iv} \right)$$

$$= \frac{1}{n} \sum_{u=1}^s \sum_{v=1}^t \hat{l}_{su}\hat{l}_{tv} \left(\sum_{i=1}^n e_{iu}e_{iv} \right)$$

$$= \sum_{u=1}^{\min(s,t)} \hat{l}_{su}\hat{l}_{tu}\hat{\sigma}_u^2 \quad (\hat{l}_{uu} = 1).$$

The last equality holds because of

$$\sum_{i=1}^n e_{iu}e_{iv} = \begin{cases} n\sigma_u^2 & u = v; \\ 0 & u \neq v. \end{cases}$$

Thus, we can establish the result

$$\mathbf{S} = \hat{\mathbf{L}} \text{diag}(\hat{\sigma}_1^2, \dots, \hat{\sigma}_p^2) \hat{\mathbf{L}}^T.$$

□

Conditional Misclassification Error of LDA

Without loss of generality, we consider a two-class classification problem here. Suppose the binary classifier function for LDA is $g(\mathbf{x}) = \log[P(Y = 1|\mathbf{X} = \mathbf{x})/P(Y = 2|\mathbf{X} = \mathbf{x})]$. Then

$$g(\mathbf{x}) \triangleq \mathbf{a}^T \mathbf{x} - b = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \left[\frac{1}{2} (\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) - \log \frac{\pi_1}{\pi_2} \right],$$

where π_1 and π_2 are the prior probabilities for class 1 and 2, respectively, i.e., $\pi_1 = P(Y = 1)$ and $\pi_2 = P(Y = 2)$. For a new observation \mathbf{x} , we predict its class $Y = 1$ if $g(\mathbf{x}) > 0$, and $Y = 2$ otherwise. Then the conditional misclassification error is

$$P(g(\mathbf{x}) = 1|Y = 2)P(Y = 2) + P(g(\mathbf{x}) = 2|Y = 1)P(Y = 1)$$

$$= P(\mathbf{a}^T \mathbf{x} - b > 0|Y = 2)\pi_2 + P(\mathbf{a}^T \mathbf{x} - b \leq 0|Y = 1)\pi_1.$$

Since $\mathbf{x}|Y = 1 \sim N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$, and $\mathbf{x}|Y = 2 \sim N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$, obviously $\mathbf{a}^T \mathbf{x}|Y = 1 \sim N(\mathbf{a}^T \boldsymbol{\mu}_1, \mathbf{a}^T \boldsymbol{\Sigma} \mathbf{a})$, and $\mathbf{a}^T \mathbf{x}|Y = 2 \sim N(\mathbf{a}^T \boldsymbol{\mu}_2, \mathbf{a}^T \boldsymbol{\Sigma} \mathbf{a})$. Therefore,

$$P(\mathbf{a}^T \mathbf{x} - b > 0|Y = 2) = \Phi \left(\frac{\mathbf{a}^T \boldsymbol{\mu}_2 - b}{\sqrt{\mathbf{a}^T \boldsymbol{\Sigma} \mathbf{a}}} \right),$$

$$P(\mathbf{a}^T \mathbf{x} - b \leq 0|Y = 1) = \Phi \left(-\frac{\mathbf{a}^T \boldsymbol{\mu}_1 - b}{\sqrt{\mathbf{a}^T \boldsymbol{\Sigma} \mathbf{a}}} \right),$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal random variable. As a result, the conditional misclassification error is

$$\pi_2 \Phi \left(\frac{\mathbf{a}^T \boldsymbol{\mu}_2 - b}{\sqrt{\mathbf{a}^T \boldsymbol{\Sigma} \mathbf{a}}} \right) + \pi_1 \Phi \left(-\frac{\mathbf{a}^T \boldsymbol{\mu}_1 - b}{\sqrt{\mathbf{a}^T \boldsymbol{\Sigma} \mathbf{a}}} \right).$$

Assume $\pi_1 = \pi_2 = 1/2$. Then with the estimates of \mathbf{a} and b through $\hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\mu}}_2, \hat{\boldsymbol{\Sigma}}$, the conditional misclassification error $\gamma(\hat{\boldsymbol{\Sigma}}, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\mu}}_2)$ is

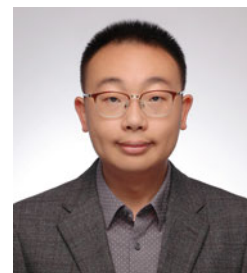
$$\gamma(\hat{\boldsymbol{\Sigma}}, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\mu}}_2)$$

$$= \frac{1}{2} \Phi \left(\frac{(\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)^T \hat{\boldsymbol{\Sigma}}^{-1} \boldsymbol{\mu}_2 - \frac{1}{2} (\hat{\boldsymbol{\mu}}_1 + \hat{\boldsymbol{\mu}}_2)^T \hat{\boldsymbol{\Sigma}}^{-1} (\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)}{\sqrt{(\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)^T \hat{\boldsymbol{\Sigma}}^{-1} \boldsymbol{\Sigma} \hat{\boldsymbol{\Sigma}}^{-1} (\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)}} \right)$$

$$+ \frac{1}{2} \Phi \left(-\frac{(\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)^T \hat{\boldsymbol{\Sigma}}^{-1} \boldsymbol{\mu}_1 - \frac{1}{2} (\hat{\boldsymbol{\mu}}_1 + \hat{\boldsymbol{\mu}}_2)^T \hat{\boldsymbol{\Sigma}}^{-1} (\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)}{\sqrt{(\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)^T \hat{\boldsymbol{\Sigma}}^{-1} \boldsymbol{\Sigma} \hat{\boldsymbol{\Sigma}}^{-1} (\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)}} \right).$$

References

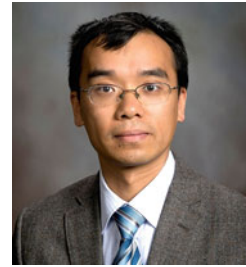
1. Bickel, P.J., Levina, E.: Covariance regularization by thresholding. *Ann. Stat.* **36**(6), 2577–2604 (2008a)
2. Bickel, P.J., Levina, E.: Regularized estimation of large covariance matrices. *Ann. Stat.* **36**(1), 199–227 (2008b)
3. Bien, J., Tibshirani, R.J.: Sparse estimation of a covariance matrix. *Biometrika* **98**(4), 807–820 (2011)
4. Cai, T.T., Yuan, M.: Adaptive covariance matrix estimation through block thresholding. *Ann. Stat.* **40**(40), 2014–2042 (2012)
5. Cai, T.T., Ren, Z., Zhou, H.H.: Estimating structured high-dimensional covariance and precision matrices: optimal rates and adaptive estimation. *Electronic Journal of Statistics* **10**(1), 1–59 (2016)
6. Chang, C., Tsay, R.S.: Estimation of covariance matrix via the sparse Cholesky factor with lasso. *J. Stat. Plann. Inference* **140**(12), 3858–3873 (2010)
7. Cochran, W.G.: *Sampling Techniques*. Wiley, New York (1977)
8. Deng, X., Lin, C.D., Liu, K.-W., Rowe, R.K.: Additive Gaussian process for computer models with qualitative and quantitative factors. *Technometrics* **59**(3), 283–292 (2017)
9. Fan, J., Li, R.: Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Am. Stat. Assoc.* **96**, 1348–1360 (2001)
10. Fan, J., Liao, Y., Liu, H.: An overview of the estimation of large covariance and precision matrices. *Econ. J.* **19**(1), 1–32 (2016)
11. Francq, C., Zakoïan, J. M.: Estimating multivariate volatility models equation by equation. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **78**(3), 613–635 (2016)
12. Friedman, J., Hastie, T., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* **33**, 1–22 (2010)
13. Huang, J.Z., Liu, N., Pourahmadi, M., Liu, L.: Covariance matrix selection and estimation via penalised normal likelihood. *Biometrika* **93**, 85–98 (2006)
14. Kang, X., Xie, C., Wang, M.: A Cholesky-based estimation for large-dimensional covariance matrices. *J. Appl. Stat.* **47**, 1017–1030 (2020)
15. Kang, X., Deng, X., Tsui, K., Pourahmadi, M.: On variable ordination of modified Cholesky decomposition for estimating time-varying covariance matrices. *Int. Stat. Rev.*, **88**(3), 616–641 (2020)
16. Lam, C., Fan, J.: Sparsistency and rates of convergence in large covariance matrix estimation. *Ann. Stat.* **37**, 4254–4278 (2009)
17. Lan, Q., Sun, H., Robertson, J., Deng, X., Jin, R.: Non-invasive assessment of liver quality in transplantation based on thermal imaging analysis. *Comput. Methods Prog. Biomed.* **164**, 31–47 (2018)
18. Leng, C., Li, B.: Forward adaptive banding for estimating large covariance matrices. *Biometrika* **98**(4), 821–830 (2011)
19. Levina, E., Zhu, R.J.: Sparse estimation of large covariance matrices via a nested lasso penalty. *Ann. Appl. Stat.* **2**(1), 245–263 (2008)
20. Liu, H., Wang, L., Zhao, T.: Sparse covariance matrix estimation with eigenvalue constraints. *J. Comput. Graph. Stat.* **23**(2), 439–459 (2014)
21. Mohammadi, A., Wit, E.C.: Bayesian structure learning in sparse Gaussian graphical models. *Bayesian Anal.* **10**(1), 109–138 (2015)
22. Nino-Ruiz, E.D., Sandu, A., Deng, X.: An ensemble Kalman filter implementation based on modified cholesky decomposition for inverse covariance matrix estimation. *SIAM J. Sci. Comput.* **40**(2), A867–CA886 (2018)
23. Pedeli, X., Fokianos, K., Pourahmadi, M.: Two Cholesky-log-GARCH models for multivariate volatilities. *Stat. Model.* **15**, 233–255 (2015)
24. Pourahmadi, M.: Joint mean-covariance models with applications to longitudinal data: unconstrained parameterisation. *Biometrika* **86**, 677–690 (1999)
25. Rajaratnam, B., Salzman, J.: Best permutation analysis. *J. Multivar. Anal.* **121**, 193–223 (2013)
26. Rigollet, P., Tsybakov, A.: Estimation of covariance matrices under sparsity constraints. *Probl. Inf. Transm.* **51**(4), 32–46 (2012)
27. Rothman, A.J., Levina, E., Zhu, J.: Generalized thresholding of large covariance matrices. *J. Am. Stat. Assoc.* **104**(485), 177–186 (2009)
28. Rothman, A.J., Levina, E., Zhu, J.: A new approach to Cholesky-based covariance regularization in high dimensions. *Biometrika* **97**(3), 539–550 (2010)
29. Sapsanis, C., Georgoulas, G., Tzes, A., Lymberopoulos, D.: Improving EMG based classification of basic hand movements using EMD. In: *Proceedings of the 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 5754–5757 (2013)
30. Sun, H., Huang, S., Jin, R.: Functional graphical models for manufacturing process modeling. *IEEE Trans. Autom. Sci. Eng.* **14**(4), 1612–1621 (2017)
31. Sun, H., Rao, P.K., Kong, Z., Deng, X., Jin, R.: Functional quantitative and qualitative models for quality modeling in a fused deposition modeling process. *IEEE Trans. Autom. Sci. Eng.* **15**(1), 393–403 (2018)
32. Tan, L.S., Nott, D.J.: Gaussian variational approximation with sparse precision matrices. *Stat. Comput.* **28**(2), 259–275 (2018)
33. Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **58**, 267–288 (1996)
34. Wagaman, A., Levina, E.: Discovering sparse covariance structures with the Isomap. *J. Comput. Graph. Stat.* **18**(3), 551–572 (2009)
35. Wu, H., Deng, X., Ramakrishnan, N.: Sparse estimation of multivariate poisson log-normal model and inverse covariance for counting data. *Stat. Anal. Data Min.* **11**, 66–77 (2018)
36. Xue, L., Ma, S., Zou, H.: Positive-definite L_1 -penalized estimation of large covariance matrices. *J. Am. Stat. Assoc.* **107**(500), 1480–1491 (2012)
37. Yu, P.L.H., Wang, X., Zhu, Y.: High dimensional covariance matrix estimation by penalizing the matrix-logarithm transformed likelihood. *Comput. Stat. Data Anal.* **114**, 12–25 (2017)
38. Yuan, M., Lin, Y.: Model selection and estimation in the Gaussian graphical model. *Biometrika* **94**, 19–35 (2007)
39. Zeng, L., Deng, X., Yang, J.: A constrained Gaussian process approach to modeling tissue-engineered scaffold degradation. *IIEE Trans.* **50**(5), 431–447 (2018)
40. Zheng, H., Tsui, K.-W., Kang, X., Deng, X.: Cholesky-based model averaging for covariance matrix estimation. *Stat. Theory Relat. Fields* **1**(1), 48–58 (2017)



Xiaoning Kang received his Ph.D. degree in statistics from Virginia Tech. He is now an associate professor at International Business College and Institute of Supply Chain Analytics in Dongbei University of Finance and Economics, China. His research interests include high-dimensional matrix estimation, mixed responses data analysis, Bayesian hierarchical modeling, and discriminant analysis.



Zhiyang Zhang is an instructor in the Department of Statistics at Virginia Tech. She received her master's degree in statistics and Ph.D. degree in chemistry from Virginia Tech. Her research interests focus on engineering statistics, data mining, and experimental design for webpage optimization.



Xinwei Deng is an associate professor in the Department of Statistics at Virginia Tech. He received his bachelor's degree in mathematics from Nanjing University and Ph.D. degree in industrial engineering from Georgia Tech. His research interests focus on statistical modeling and data analysis, including high-dimensional classification, graphical model estimation, and the interface between experimental design and machine learning. He is an elected member of ISI and a member of INFORMS and ASA.



Contents

44.1	What Is Statistical Learning	901
44.1.1	Literature Review.....	902
44.2	Uniform Convergence	903
44.2.1	Concentration Inequalities.....	903
44.2.2	Rademacher Complexity.....	904
44.2.3	Growth Function and VC Dimension.....	905
44.2.4	Covering Number.....	906
44.3	Learning Theory for Deep Learning	907
44.4	Learning Theory for Robust Adversarial Learning	909
44.5	Example: Generalization Bounds for Logistic Regression	911
44.6	Discussion	912
	Appendix	912
	References	918

Abstract

One of the main goals of statistical learning is to characterize how the excess risk depends on the sample size n , on the complexity of the hypothesis class, and on the underlying complexity of the prediction problem itself. A related problem is to control the generalization error, which is a measure of how accurately an algorithm is able to predict outcome values for previously unseen data. Establishing probability error bounds for these problems can be converted into a problem of uniform convergence. We first introduce some commonly used technical tools for uniform convergence. Along the way, we highlight the recent development of learning theory for deep neural networks (DNNs) and explain the theoretical benefit to improve the generalization error in practice. Furthermore, we

present the generalization of DNNs for robust adversarial learning with ℓ_∞ attacks. For general machine learning tasks, we show that adversarial Rademacher complexity is always larger than natural counterpart, but the effect of adversarial perturbations can be limited under the weight normalization framework.

Keywords

Adversarial learning · Excess risk · Generalization error · Learning theory · Rademacher complexity · Uniform convergence

44.1 What Is Statistical Learning

In supervised learning problems such as classification and regression, our target is to predict an output $y \in \mathcal{Y}$ based on a set of features $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$. Informally, we choose a predictor $f : \mathcal{X} \rightarrow \mathcal{Y}$ from the hypothesis class \mathcal{F} such that $f(\mathbf{x})$ is a good prediction of y . Let (\mathbf{x}, y) be from an unknown distribution \mathcal{D} and the loss function be $\ell(\cdot, \cdot) : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$. Define the *expected risk* as $R(f) = \mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}} [\ell(f(\mathbf{x}), y)]$, and our goal is to find the expected risk minimizer which is denoted by $f^* \in \arg \min_{f \in \mathcal{F}} R(f)$. Given n i.i.d. samples $S = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$, where each pair is from \mathcal{D} over $\mathcal{X} \times \mathcal{Y}$, the approximation of f^* is obtained by minimizing the *empirical risk*:

$$\widehat{R}(f) = \frac{1}{n} \sum_{i=1}^n \ell(f(\mathbf{x}_i), y_i). \quad (44.1)$$

The trained predictor f is also called the *empirical risk minimizer* (ERM) defined as any hypothesis $f \in \mathcal{F}$ that minimizes Eq. (44.1):

$$\widehat{f} \in \arg \min_{f \in \mathcal{F}} \widehat{R}(f). \quad (44.2)$$

Q. Gao · X. Wang (✉)
Purdue University, West Lafayette, IN, USA
e-mail: gao424@purdue.edu; wangxiao@purdue.edu

In practice, we often choose the quadratic loss for regression problems, while the hinge loss and the cross entropy loss are commonly used in classification tasks. Statistical learning is an active area of research in the past two decades: well-known monographs in this area include [8, 26, 37].

The key question of statistical learning is to analyze and control the *excess risk*, which is the difference between $R(f^*)$ and $R(\hat{f})$. The excess risk characterizes the gap between the expected risk of \hat{f} and the optimal f^* . Another related concept is called the *generalization error*, which is the difference between $R(\hat{f})$ and $\widehat{R}(\hat{f})$. Mathematically, the generalization error is a measure of how accurately an algorithm is able to predict outcome values for previously unseen data. The generalization error can be minimized by avoiding overfitting in the learning algorithm. We will show later that the generalization error is easy to control if the excess risk is bounded. So, how do we analyze the excess risk? Note that the excess risk is a random variable depending on the training set via \hat{f} , and the sample size n is finite. Therefore, the central limit theorem in asymptotics cannot be directly applied here. We formulate the analysis as a probability statement. Given $\eta \in (0, 1)$, the excess risk is upper bounded by some ϵ with probability at least $1 - \eta$, that is, $\mathbb{P}(R(\hat{f}) - R(f^*) \leq \epsilon) \geq 1 - \eta$, or equivalently,

$$\mathbb{P}(R(\hat{f}) - R(f^*) \geq \epsilon) \leq \eta, \quad (44.3)$$

where ϵ is a function relying on η and the complexity of the hypothesis class \mathcal{F} .

To explicitly describe ϵ , we rewrite the excess risk as

$$R(\hat{f}) - R(f^*) = \underbrace{R(\hat{f}) - \widehat{R}(\hat{f})}_{(a)} + \underbrace{\widehat{R}(\hat{f}) - \widehat{R}(f^*)}_{(b)} + \underbrace{\widehat{R}(f^*) - R(f^*)}_{(c)}.$$

Term (b), $\widehat{R}(\hat{f}) - \widehat{R}(f^*)$, is nonpositive, because \hat{f} is chosen to minimize $\widehat{R}(f)$ in the hypothesis class \mathcal{F} . Term (c) is the difference between a sample average and an expectation in terms of the fixed function f^* , such that

$$\widehat{R}(f^*) - R(f^*) = \frac{1}{n} \sum_{i=1}^n \ell(f^*(\mathbf{x}_i), y_i) - \mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}} [\ell(f^*(\mathbf{x}), y)].$$

The law of large numbers shows that this term converges to zero. With information about the tails of $\ell(f^*(\mathbf{x}), y)$ such as boundedness, we can use concentration inequalities that we will introduce in the next section to bound its value. Term (a), $R(\hat{f}) - \widehat{R}(\hat{f})$, is more interesting and complicated, since \hat{f} is random based on the chosen data. An easy approach is to provide a uniform upper bound,

$$R(\hat{f}) - \widehat{R}(\hat{f}) \leq \sup_{f \in \mathcal{F}} |R(f) - \widehat{R}(f)|,$$

which motivates us to study the uniform convergence. Suppose we can ensure that $R(f)$ and $\widehat{R}(f)$ were close (say within $\epsilon/2$) for all $f \in \mathcal{F}$. Then, we could guarantee that $R(\hat{f})$ and $\widehat{R}(\hat{f})$ were within $\epsilon/2$, as well as $R(f^*)$ and $\widehat{R}(f^*)$. Therefore, Eq. (44.3) can be written formally as

$$\mathbb{P}(R(\hat{f}) - R(f^*) \geq \epsilon) \leq \mathbb{P}\left(\sup_{f \in \mathcal{F}} |R(f) - \widehat{R}(f)| \geq \frac{\epsilon}{2}\right). \quad (44.4)$$

On the right-hand side is a statement about uniform convergence, which describes the probability of the event that the largest difference between the empirical and expected risk is at least $\epsilon/2$, or equivalently, the event that this difference exceeds $\epsilon/2$ for at least one $f \in \mathcal{F}$.

44.1.1 Literature Review

Statistical learning has been widely used to analyze the generalization performance such as kernel methods, support vector machine (SVM), and deep neural networks (DNNs).

The good generalization ability of SVM is usually explained by the existence of a large margin. Such bounds on the error rate for a hyperplane that separates the data with some margin were obtained in [3, 31], which are mainly based on the VC dimension. Reference [38] showed that the generalization ability of SVM depends on more complex geometrical constructions than large margin by proposing a new concept called the span of support vectors. Kernel methods have been successfully used in SVM, and the choice of kernel is very critical. There is a large body of literature dealing with the problem of learning kernels. For example, [33] gave generalization bounds for linear combinations of kernels with L_1 regularization that have an order of $\mathcal{O}(\sqrt{(p + 1/\rho^2)/n})$, where p is the number of kernels combined, n is the sample size, and ρ is the margin of the learned classifier. Cortes et al. [10] presented new generalization bounds for the family of convex combinations of base kernels by using the Rademacher complexity of the hypothesis set, which is of order $\mathcal{O}(\sqrt{(\log p)/(\rho^2 n)})$.

The generalization bound of DNNs has been extensively studied in literature, especially norm-constrained fully connected DNNs [4, 6, 13, 22, 27, 28, 34]. In particular, spectral norm-constrained fully connected DNNs were studied in [6, 22, 27]. Assume that the spectral norm of the weight matrix in each layer equals to 1, and the width of each hidden layer is d . Then the corresponding bound of generalization error is of order $\sqrt{d^3 L^2/n}$ [6, 27] and $\sqrt{dLr/n}$ [22], respectively,

where n is the sample size, L is the depth, and r is the rank of weight matrices. On the other hand, a lower bounds for the generalization error with an order of $\sqrt{d/n}$ is established in [13]. In addition, some special cases of the matrix mixed $L_{p,q}$ norm-constrained fully connected DNNs were studied in [4, 13, 28, 34, 39]. For example, [28] provided an exponential bound on the width d based on the Frobenius norm of the weight matrices; [6] provided a polynomial bound on L and d based on the spectral norm and the $L_{2,1}$ norm.

However, many recent studies have shown that DNNs are vulnerable to adversarial attacks [7, 11, 36], which means DNNs may not predict correct labels for inputs with small adversarial perturbations. Reference [30] showed that for natural learning, $\mathcal{O}(1)$ training data is enough to correctly classify two separated d -dimensional Gaussian distribution, but $\Theta(\sqrt{d})$ data is needed for adversarial learning. Reference [2] provided adversarial generalization bounds only considering a finite number of potential adversarial attacks. Reference [40] used the SDP relaxation proposed by [29] to establish adversarial generalization bound, but their results are limited to linear classifiers and single hidden layer neural networks. Reference [17] used a surrogate adversarial loss called tree transform to upper bound the actual adversarial loss and reduced the problem of deriving bounds on the adversarial risk to the problem of deriving risk bounds using standard learning-theoretic techniques. However, their bounds rely on ℓ_∞ -operator norms, Frobenius norms, and $L_{p,\infty}$ norms of weight matrices, and polynomially depend on the depth of DNNs. Reference [12] extended PAC-Bayes framework to bound the generalization error for DNNs under several particular adversarial strategies, but they focused on ℓ_2 perturbations rather than ℓ_∞ perturbations, and their results are based on spectral norms and Frobenius norms.

44.2 Uniform Convergence

In this section, we introduce some commonly used techniques for establishing uniform convergence.

44.2.1 Concentration Inequalities

Let us first recall the central limit theorem. Assume X_1, \dots, X_n are n independent random variables with $\mathbb{E}X_i = \mu$ and $\text{Var}X_i = \sigma^2 < \infty$, then

$$\lim_{n \rightarrow \infty} \mathbb{P}(\bar{X}_n \geq \mu + \sigma t / \sqrt{n}) = 1 - \Phi(t) \quad (44.5)$$

The central limit theorem tells us what happens asymptotically, but we usually have a fixed sample size. In this case, what is $\mathbb{P}(|\bar{X}_n - \mu| \geq \epsilon)$? It is answered by concentration inequalities, a very powerful set of techniques providing

bounds on this kind of probability that \bar{X}_n is concentrated about its mean. We start from Markov's inequality, a simple tool that allows us to control the deviation of a nonnegative random variable from its mean using the expectation of that random variable.

Theorem 44.2.1 (Markov's Inequality) For a random variable $X \geq 0$ a.s., $\mathbb{E}X < \infty$, $t > 0$: $\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}X}{t}$.

We use $|X - \mathbb{E}X|$ to replace X in Markov's inequality and consider a strictly monotonic function $f : [0, \infty) \mapsto [0, \infty)$. Assume $\mathbb{E}[f(|X - \mathbb{E}X|)] < \infty$, we have

$$\begin{aligned} \mathbb{P}(|X - \mathbb{E}X| \geq t) &= \mathbb{P}(f(|X - \mathbb{E}X|) \geq f(t)) \\ &\leq \frac{\mathbb{E}[f(|X - \mathbb{E}X|)]}{f(t)} \end{aligned}$$

Different choices of f lead to different inequalities. $f(a) = a^2$ gives Chebyshev's inequality

$$\mathbb{P}(|X - \mathbb{E}X| \geq t) \leq \frac{\text{Var}(X)}{t^2}.$$

When $f(a) = a^k$, we have

$$\mathbb{P}(|X - \mathbb{E}X| \geq t) \leq \frac{\mathbb{E}|X - \mathbb{E}X|^k}{t^k}.$$

When $f(a) = \exp(\lambda a)$ for $\lambda > 0$, we have the Chernoff inequality

$$\begin{aligned} \mathbb{P}(X - \mathbb{E}X \geq t) &= \mathbb{P}(\exp(\lambda(X - \mathbb{E}X)) \geq \exp(\lambda t)) \\ &\leq e^{-\lambda t} M_{X-\mu}(\lambda), \end{aligned} \quad (44.6)$$

where $\mathbb{E}X = \mu$, and $M_{X-\mu}(\lambda) = \mathbb{E} \exp(\lambda(X - \mu))$ is the moment generating function. Consider a special case of Gaussian distribution where $X \sim N(\mu, \sigma^2)$, then $M_{X-\mu}(\lambda) = \exp(\lambda^2 \sigma^2 / 2)$. Thus,

$$\mathbb{P}(X - \mu \geq t) \leq \inf_{\lambda} \exp\left(\frac{\lambda^2 \sigma^2}{2} - \lambda t\right) \leq \exp\left(-\frac{t^2}{2\sigma^2}\right),$$

using the optimal choice $\lambda = t/\sigma^2 > 0$. How about non-Gaussian variables? Note that the bounds would still hold as long as $M_{X-\mu}(\lambda)$ is upper bounded. This leads to the definition of sub-Gaussian.

Definition 44.2.1 X is sub-Gaussian with parameter σ^2 if, for all $\lambda \in \mathbb{R}$,

$$M_{X-\mu}(\lambda) \leq \exp\left(\frac{\lambda^2 \sigma^2}{2}\right).$$

If X is a Gaussian random variable from $N(0, \sigma^2)$, then X is sub-Gaussian with parameter σ^2 . However, exponential and Gamma variables are not sub-Gaussian, since the tails of these distributions are too fat. There are some properties of sub-Gaussian.

- X sub-Gaussian iff $-X$ sub-Gaussian.
- If X_1, X_2 are independent sub-Gaussian with parameters σ_1^2, σ_2^2 , then $X_1 + X_2$ is sub-Gaussian with parameter $\sigma_1^2 + \sigma_2^2$.
- If X is sub-Gaussian with parameter σ^2 , then for any $c > 0$, cX is sub-Gaussian with parameter $c^2\sigma^2$.

Equipped with this machinery, we can easily obtain the following classic tail bound for bounded random variables.

Theorem 44.2.2 For independent X_1, \dots, X_n , $\mathbb{E}X_i = \mu_i$, and X_i is sub-Gaussian with parameter σ_i^2 , then for all $t > 0$,

$$\mathbb{P}\left(\sum_{i=1}^n (X_i - \mu_i) \geq t\right) \leq \exp\left(-\frac{t^2}{2\sum_{i=1}^n \sigma_i^2}\right).$$

The proof is easy by applying the Chernoff inequality. If each X_i is bounded in $[a_i, b_i]$, then $\sigma_i^2 = (b_i - a_i)^2/4$. Thus, we have Hoeffding’s inequality

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^n X_i - \frac{1}{n}\sum_{i=1}^n \mu_i \geq t\right) \leq \exp\left(-\frac{2n^2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\right).$$

Another important tool is the McDiarmid’s inequality, which is a generalization of the Hoeffding’s inequality, where we want to bound not the average of random variables $X_1 \dots, X_n$ but any function on X_1, \dots, X_n satisfying an appropriate bounded differences condition.

Theorem 44.2.3 (McDiarmid’s Inequality) Let X_1, \dots, X_n be independent random variables with support on \mathcal{X} . Let $f : \mathcal{X}^n \mapsto \mathbb{R}$ be a function satisfying the following bounded difference condition,

$$(\forall i, \forall x_1, \dots, x_n, x'_i) \quad \left|f(x_1, \dots, x_i, \dots, x_n) - f(x_1, \dots, x'_i, \dots, x_n)\right| \leq B_i,$$

then,

$$\begin{aligned} \mathbb{P}(|f(X_1, \dots, X_n) - \mathbb{E}[f(X_1, \dots, X_n)]| \geq t) \\ \leq 2 \exp\left(-\frac{2t^2}{\sum_i B_i^2}\right). \end{aligned}$$

We apply martingale to prove this inequality, and please refer to [26] for more details. This is a quite powerful result, as

it holds for any independent random variables, even if f is complex such as neural networks. As long as the function is not too sensitive to perturbations in one of its arguments, we get good concentration.

So far, we have introduced several main concentration inequalities. To have a better understanding, we now use them to analyze the generalization results for a finite hypothesis class. This is accomplished by a two-step concentration and the union bound.

Theorem 44.2.4 Let $S = \{(x_1, y_1), \dots, (x_n, y_n)\}$ be n i.i.d. samples from the unknown distribution \mathcal{D} . Assume that \mathcal{F} is a finite hypothesis class, i.e., $\mathcal{F} = \{f_1, \dots, f_k\}$, where $k < \infty$ and $f_j : \mathcal{X} \mapsto \mathcal{Y}$ for $\forall j$. Let ℓ be the zero-one loss, i.e., $\ell(f(\mathbf{x}), y) = \mathbb{I}[f(\mathbf{x}) \neq y]$. Let \hat{f} be the ERM defined in Eq. (44.2). For fix $\eta \in (0, 1)$, with probability at least $1 - \eta$, we have

$$R(\hat{f}) - R(f^*) \leq \sqrt{\frac{2(\log k + \log(2/\eta))}{n}}.$$

44.2.2 Rademacher Complexity

In the previous section, we analyzed the excess risk with a finite hypothesis class \mathcal{F} , i.e., $|\mathcal{F}| < \infty$. However, the union bound cannot be applied to infinite hypothesis classes. This motivates us to explore more sophisticated approaches to measure the capacity of a hypothesis class. In this section, we will introduce a framework called *Rademacher complexity* to uniformly bound the difference between the expected and empirical risk for any $f \in \mathcal{F}$.

Definition 44.2.2 (Empirical Rademacher Complexity)

Let \mathcal{H} be a class of real-valued functions $h : \mathcal{Z} \mapsto \mathbb{R}$. The empirical Rademacher complexity of \mathcal{H} is defined as

$$\widehat{\mathfrak{R}}_n(\mathcal{H}) = \mathbb{E}_\delta \left[\sup_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^n \delta_i h(z_i) \right],$$

where $S = \{z_1, \dots, z_n\}$ includes n i.i.d. samples from distribution \mathcal{D} and $\delta = \{\delta_1, \dots, \delta_n\}$ are n i.i.d. Rademacher variables satisfying $\mathbb{P}(\delta_i = 1) = \mathbb{P}(\delta_i = -1) = 1/2$.

It is clear that the empirical Rademacher complexity is a random variable depending on the data. By further taking exception on these random samples, we have the definition of Rademacher complexity.

Definition 44.2.3 (Rademacher Complexity)

Let \mathcal{H} be a class of real-valued functions $h : \mathcal{Z} \mapsto \mathbb{R}$. The Rademacher complexity of \mathcal{H} is defined as $\mathfrak{R}_n(\mathcal{H}) = \mathbb{E}_{S \sim \mathcal{D}^n} [\widehat{\mathfrak{R}}_n(\mathcal{H})]$.

Here we give an intuitive explanation about the Rademacher complexity. Consider the simple binary classification problem with inputs z_1, \dots, z_n . If the corresponding labels are random $\delta_1, \dots, \delta_n$, this becomes a meaningless learning problem. Therefore, the Rademacher complexity is used to capture how well the best function from the function class \mathcal{H} can fit these random labels. A large \mathcal{H} will be able to fit noise better and thus have a larger Rademacher complexity. As we will see later, we would like $\mathfrak{R}_n(\mathcal{H})$ to go to zero as n increases.

The basic properties of Rademacher complexity are listed as follows:

- $\mathfrak{R}_n(\mathcal{H}) = 0$ for $\mathcal{H} = \{h\}$.
- $\mathfrak{R}_n(\mathcal{H}_1) \leq \mathfrak{R}_n(\mathcal{H}_2)$ if $\mathcal{H}_1 \subseteq \mathcal{H}_2$.
- $\mathfrak{R}_n(\mathcal{H}_1 + \mathcal{H}_2) \leq \mathfrak{R}_n(\mathcal{H}_1) + \mathfrak{R}_n(\mathcal{H}_2)$ for $\mathcal{H}_1 + \mathcal{H}_2 = \{h_1 + h_2 : h_1 \in \mathcal{H}_1, h_2 \in \mathcal{H}_2\}$
- $\mathfrak{R}_n(c\mathcal{H}) = |c|\mathfrak{R}_n(\mathcal{H})$
- $\mathfrak{R}_n(\text{co}(\mathcal{H})) = \mathfrak{R}_n(\mathcal{H})$, where $\text{co}(\mathcal{H})$ is the convex hull of \mathcal{H} .
- $\mathfrak{R}_n(\phi \circ \mathcal{H}) \leq c_\phi \mathfrak{R}_n(\mathcal{H})$, where $\phi \circ \mathcal{H} = \{z \mapsto \phi(h(z)) : h \in \mathcal{H}\}$ and c_ϕ is the Lipschitz constant of ϕ . This is also called Ledoux-Talagrand contraction inequality (see [21] for detailed proof).

The Ledoux-Talagrand contraction inequality is quite useful when analyzing the Rademacher complexity of loss class $\ell \circ \mathcal{F}$, since we can transfer it to analyze the complexity of our hypothesis class \mathcal{F} . Next, we will show the crucial theorem that links the uniform convergency and Rademacher complexity.

Theorem 44.2.5 *Let $S = \{(x_1, y_1), \dots, (x_n, y_n)\}$ be n i.i.d. samples drawn from the unknown distribution \mathcal{D} . Let \mathcal{F} be a hypothesis class and ℓ be the loss function where $\ell \circ \mathcal{F}$ belongs to $\{\ell(f(\mathbf{x}), y) \mid \ell \circ f : \mathcal{X} \times \mathcal{Y} \rightarrow [0, 1], f \in \mathcal{F}\}$. Fix $\eta \in (0, 1)$. With probability at least $1 - \eta$, we have*

$$(\forall f \in \mathcal{F}) R(f) \leq \widehat{R}(f) + 2\mathfrak{R}_n(\ell \circ \mathcal{F}) + \sqrt{\frac{\log(1/\eta)}{2n}}.$$

Equipped with this theorem, we further need several steps to bound Eq. (44.3). We analogously define $G'(S)$ for the negative loss $\ell'(f(\mathbf{x}), y) = -\ell(f(\mathbf{x}), y)$, so

$$\mathbb{P}\left(R(\widehat{f}) - R(h^*) \geq \epsilon\right) \leq \mathbb{P}\left(\sup_{f \in \mathcal{F}} |R(f) - \widehat{R}(f)| \geq \frac{\epsilon}{2}\right) \leq \mathbb{P}\left(G(S) \geq \frac{\epsilon}{2}\right) + \mathbb{P}\left(G'(S) \geq \frac{\epsilon}{2}\right).$$

Note that by the proof of Theorem 44.2.5 (see [5]), $\mathbb{E}[G(S)] \leq 2\mathfrak{R}_n(\ell \circ \mathcal{F})$ and

$$\mathbb{P}\left(G(S) - \mathbb{E}[G(S)] \geq \epsilon\right) \leq \exp(-2n\epsilon^2).$$

Since $\mathfrak{R}_n(\ell \circ \mathcal{F}) = \mathfrak{R}_n(-\ell \circ \mathcal{F})$, we have $\mathbb{E}[G'(S)] \leq 2\mathfrak{R}_n(\ell \circ \mathcal{F})$. Therefore,

$$\begin{aligned} \mathbb{P}\left(G(S) \geq \frac{\epsilon}{2}\right) &\leq \exp\left(-2n\left(\frac{\epsilon}{2} - \mathbb{E}[G(S)]\right)^2\right) \\ &\leq \exp\left(-2n\left(\frac{\epsilon}{2} - 2\mathfrak{R}_n(\ell \circ \mathcal{F})\right)^2\right) \\ &\quad \text{for } \epsilon \geq 4\mathfrak{R}_n(\ell \circ \mathcal{F}) \\ &\stackrel{\text{def}}{=} \frac{\eta}{2} \end{aligned}$$

Similarly, we have $\mathbb{P}\left(G(S) \geq \frac{\epsilon}{2}\right) \leq \frac{\eta}{2}$. Adding them together and solving for ϵ , with probability at least $1 - \eta$, we have

$$R(\widehat{f}) - R(f^*) \leq 4\mathfrak{R}_n(\ell \circ \mathcal{F}) + \sqrt{\frac{2 \log(2/\eta)}{n}}.$$

Using this conclusion and the Ledoux-Talagrand contraction lemma, we can easily bound the difference between empirical risk and expected risk as well as the excess risk once we have the Rademacher complexity of hypothesis class \mathcal{F} . There are several tools helping control $\mathfrak{R}_n(\mathcal{F})$ such as VC dimension and covering number that we will see later.

Next, we provide a simple example of how to upper bound the empirical Rademacher complexity for a class of linear predictors.

Example 44.2.1 Let $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ be n i.i.d. samples from $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\|_2 \leq B\}$. Let \mathcal{F} be the class of linear predictors, i.e.,

$$\mathcal{F} = \{(\mathbf{w}, \mathbf{x}) \mid \mathbf{w} \in \mathbb{R}^d \text{ and } \|\mathbf{w}\|_2 \leq c\}.$$

We have $\widehat{\mathfrak{R}}_n(\mathcal{F}) \leq \frac{Bc}{\sqrt{n}}$.

44.2.3 Growth Function and VC Dimension

So far, we have set up Rademacher complexity as a measure of the capacity of infinite hypothesis class. Let us instantiate Rademacher complexity when the function class has finite possible outputs such as binary classification problem. Assume the dataset $S = \{z_1, \dots, z_n\}$ contains n i.i.d. samples from distribution \mathcal{D} . In general, we assume a function class $\mathcal{H} \subseteq \{h|h : \mathcal{Z} \rightarrow \{0, 1\}\}$. We introduce the following shorthand notation: $\mathcal{H}(S) = \{(h(z_1), \dots, h(z_n)) \in \{0, 1\}^n \mid h \in \mathcal{H}\}$. That is, $\mathcal{H}(S)$ contains all the $\{0, 1\}^n$ vectors that can be produced by applying all functions in \mathcal{H} to the dataset S .

Definition 44.2.4 (Growth Function) The growth function (or shatter coefficient) of a class of functions $\mathcal{H} \subseteq \{h|h : \mathcal{Z} \rightarrow \{0, 1\}\}$ for n samples is

$$G(\mathcal{H}, n) = \max_{S \in \mathcal{Z}^n} |\mathcal{H}(S)|.$$

For example, for the class of functions $\mathcal{H} = \{z \mapsto \mathbb{I}[z \leq \alpha] : z \in \mathbb{R}, \alpha \in \mathbb{R}\}$, we have $|\mathcal{H}(S)| \leq G(\mathcal{H}, n) = n + 1$. Meanwhile, for boolean functions, if $G(\mathcal{H}, n) = 2^n$, meaning we obtain all possible labels, we say \mathcal{H} shatters any n points z_1, \dots, z_n that achieve the maximum of $\mathcal{H}(S)$. One advantage of growth function is that it turns the infinite function class to a finite coefficient. Therefore, we can directly use the following Massart’s finite lemma to link with Rademacher complexity.

Lemma 44.2.1 (Massart’s Finite Lemma) For $\mathcal{A} \subseteq \mathbb{R}^n$ with $R^2 = \frac{\max_{a \in \mathcal{A}} \|a\|_2^2}{n}$,

$$\mathbb{E} \left[\sup_{a \in \mathcal{A}} \frac{1}{n} \sum_{i=1}^n \delta_i a_i \right] \leq \sqrt{\frac{2R^2 \log |\mathcal{A}|}{n}},$$

where $\delta = \{\delta_1, \dots, \delta_n\}$ are n i.i.d. Rademacher random variables.

Taking $\mathcal{A} = \mathcal{H}(S)$, we have $R^2 \leq 1$. By Massart’s finite lemma, it is straightforward that

$$\widehat{\mathfrak{R}}_n(\mathcal{H}) \leq \sqrt{\frac{2 \log G(\mathcal{H}, n)}{n}}.$$

Thus, to get meaningful bounds, we want $G(\mathcal{H}, n)$ to grow sub-exponentially with n . Otherwise, the Rademacher complexity will not go to zero, and we will not obtain uniform convergence. This is expected since if \mathcal{H} can really hit all labels for all n , we would be able to fit any label of the data, leading to massive overfitting.

Although the growth function nicely captures the behavior of an infinite \mathcal{H} , it is not necessarily the most convenient quantity to get a handle on. In the following, we use a concept called *VC dimension* to gain more intuition about the growth function.

Definition 44.2.5 The VC dimension of a class of functions \mathcal{H} with Boolean outputs is the maximum number of points that can be shattered by \mathcal{H} :

$$VC(\mathcal{H}) = \max_{n \in \mathbb{N}} \{n | G(\mathcal{H}, n) = 2^n\}$$

Here we list several examples of VC dimension.

- For $\mathcal{H} = \{z \mapsto \mathbb{I}[z \leq \alpha] : \alpha \in \mathbb{R}\}$, $VC(\mathcal{H}) = 1$.
- For $\mathcal{H} = \{z \mapsto \mathbb{I}[z \text{ below and to left of } y] : y \in \mathbb{R}^2\}$, $VC(\mathcal{H}) = 2$.

- For $\mathcal{H} = \{z \mapsto \mathbb{I}[z \text{ in box defined by } a, b] : a, b \in \mathbb{R}^d\}$, $VC(\mathcal{H}) = 2d$
- For $\mathcal{H} = \{x \mapsto \mathbb{I}[\sin(\theta x) \geq 0] : \theta \in \mathbb{R}\}$, $VC(\mathcal{H}) = \infty$.

Theorem 44.2.6 For the class of thresholded linear functions,

$$\mathcal{H} = \{z \mapsto \mathbb{I}[f(x) \geq 0] : f \in \mathcal{F}\},$$

where \mathcal{F} is a linear space, $VC(\mathcal{H}) = \dim(\mathcal{F})$.

Lemma 44.2.2 (Sauer-Shelah Lemma) For a function class \mathcal{H} with Boolean outputs and VC dimension d , then we have

$$G(\mathcal{H}, n) \leq \sum_{i=0}^d \binom{n}{i} \leq (n+1)^d.$$

This proof of Sauer-Shelah Lemma is very technical. For more details please refer to [26]. Combining this theorem with the previous conclusion, we have

$$\widehat{\mathfrak{R}}_n(\mathcal{H}) \leq \sqrt{\frac{2 \log G(\mathcal{H}, n)}{n}} \leq \sqrt{\frac{2VC(\mathcal{H}) \log(n+1)}{n}}.$$

44.2.4 Covering Number

For infinite hypothesis classes, we observe that growth function and VC dimension are appropriate measures since all that mattered was the behavior of a function class on a finite set of points. However, these two approaches only work for functions that return a finite number of values. Can we retain the combinatorial nature of growth function but allow for real-valued functions such as regression problems? We explore covering numbers in the section to solve this problem. Covering numbers count the number of balls of size ϵ one needs to cover the hypothesis class, then the Massart’s finite lemma can be applied to control the representatives. In essence, covering numbers allow us to discretize the problem.

Definition 44.2.6 A metric space (\mathcal{X}, ρ) is a set \mathcal{X} and a function $\rho : \mathcal{X} \times \mathcal{X} \mapsto [0, \infty)$ satisfying

- Identity of indiscernibles: $\rho(x, x) = 0$
- Symmetry: $\rho(x, y) = \rho(y, x)$
- Triangle inequality: $\rho(x, z) \leq \rho(x, y) + \rho(y, z)$

If $\rho(x, y) = 0$ is possible for $x \neq y$, then we say ρ is a pseudometric. In this section, we will work with the pseudometric. For example, the pseudometric for a set of functions \mathcal{H} mapping from \mathcal{Z} to \mathbb{R} is

$$\rho(h, h') = \|h - h'\|_{L_1(P_n)} := \frac{1}{n} \sum_{i=1}^n |h(z_i) - h'(z_i)|.$$

Definition 44.2.7

- An ϵ -cover of a subset \mathcal{H} of a pseudometric space (\mathcal{X}, ρ) is a finite set $\mathcal{H}' \subset \mathcal{H}$ such that for each $h \in \mathcal{H}$ there is a $h' \in \mathcal{H}'$ such that $\rho(h, h') \leq \epsilon$.
- The ϵ -covering number of \mathcal{H} is

$$\mathcal{N}(\epsilon, \mathcal{H}, \rho) = \min\{|\mathcal{H}'| : \mathcal{H}' \text{ is an } \epsilon\text{-cover of } \mathcal{H}\}.$$

- $\log \mathcal{N}(\epsilon, \mathcal{H}, \rho)$ is the metric entropy of \mathcal{H} .
- If $\lim_{\epsilon \rightarrow 0} \log \mathcal{N}(\epsilon) / \log(1/\epsilon)$ exists, it is called the metric dimension.

From above definitions, it is straightforward that as ϵ decreases, h' in the cover \mathcal{H}' is a better approximation of h , but $\mathcal{N}(\epsilon, \mathcal{H}, \rho)$ also increases. In general, we would like $\mathcal{N}(\epsilon, \mathcal{H}, \rho)$ to be small, so what is the trade-off? The following theorems establish that the covering number enables to upper bound the Rademacher complexity, which also provide hints for this trade-off.

Theorem 44.2.7 (Discretization Theorem) *Let $S = \{z_1, \dots, z_n\} \in \mathcal{Z}$ be n i.i.d. samples from distribution \mathcal{D} . Let \mathcal{H} be a class of functions $h : \mathcal{Z} \mapsto [-1, 1]$, and consider the $L_2(P_n)$ pseudometric on \mathcal{H} ,*

$$\rho(h, h') = \left[\frac{1}{n} \sum_{i=1}^n (h(z_i) - h'(z_i))^2 \right]^{1/2}.$$

Then,

$$\widehat{\mathfrak{R}}_n(\mathcal{H}) \leq \inf_{\epsilon > 0} \left(\sqrt{\frac{2 \log(2\mathcal{N}(\epsilon, \mathcal{H}, \rho))}{n}} + \epsilon \right).$$

The first term on the right-hand side is the covering number, which increases as ϵ decreases as we discussed. The second term is ϵ , which can be regarded as the penalty we pay for having a discretization. We now introduce a clever technique called chaining gets at the intuition by constructing covers at multiple levels of resolution and using functions in the coarser covers as waypoints. The idea is that the fewer functions in the coarse cover can have larger deviation, and the many functions in the finer covers has smaller deviation. Note that compared with the discretization theorem, this bound involves an integral that sweeps across different resolutions ϵ and importantly removes the additive ϵ penalty.

Theorem 44.2.8 (Dudley's Theorem) *Let \mathcal{H} be a family of functions $h : \mathcal{Z} \mapsto \mathbb{R}$. Consider the $L_2(P_n)$ pseudometric on \mathcal{H} , then*

$$\widehat{\mathfrak{R}}_n(\mathcal{H}) \leq 12 \int_0^\infty \sqrt{\frac{2 \log \mathcal{N}(\epsilon, \mathcal{H}, \rho)}{n}} d\epsilon.$$

In simple discretization, we apply Massart's finite class lemma on functions whose magnitude is $\sup_{h \in \mathcal{H}} \|h\|_{L_2(P_n)}$, whereas in chaining, we apply this lemma on differences between functions in \mathcal{H}'_j and \mathcal{H}'_{j-1} whose range is $3\epsilon_j$. This leads chaining to produce better results.

44.3 Learning Theory for Deep Learning

In this section, we will mainly summarize recent results on the generalization error of deep neural networks (DNNs). DNNs are the essence of deep learning models. The goal of a DNN is to approximate the unknown true function. Reference [16] stated that a neural network with at least one hidden layer with any "squashing" activation function (such as the logistic sigmoid activation function) can approximate any Borel measurable function with any desired nonzero amount of error, provided that the network is given enough hidden units. This universal property makes it the basis of many important commercial applications. For instance, the convolutional neural networks used for object recognition from photos are a specialized kind of neural network. It is also a conceptual stepping stone on the path to recurrent networks, which power many natural language applications.

The architecture of neural networks is composed of an input layer, multiple hidden layers, and an output layer. We use $\mathcal{F}_{\mathcal{W}_L}$ to denote a class of neural networks. Given the input domain $\mathcal{X} = \{\mathbf{x} : \mathbf{x} \in \mathbb{R}^{d_0}\}$, the form of each $f_{\mathcal{W}_L} \in \mathcal{F}_{\mathcal{W}_L}$ is

$$f_{\mathcal{W}_L}(\mathbf{x}) = \sigma_L(W_L \sigma_{L-1}(W_{L-1} \cdots \sigma_1(W_1 \mathbf{x}))), \quad (44.7)$$

where for $i = 1, \dots, L$, W_i is the $\mathbb{R}^{d_i} \times \mathbb{R}^{d_{i-1}}$ weight matrix, and $\sigma_i(\cdot)$ is element-wise nonlinear activation functions satisfying $\sigma_i(\mathbf{0}) = \mathbf{0}$. In the above, we use \mathcal{W}_L to denote all weight matrices (W_1, \dots, W_L) . We denote L as the depth and $d = \max\{d_0, \dots, d_L\}$ as the width of the neural network, which are the total number of layers and the maximum number of neurons in each layer, respectively. Typical examples of activation functions include sigmoid, tanh, and Relu. Note that Relu owns good properties, satisfying $\sigma(0) = 0$ and 1-Lipschitz contentiousness.

Existing results imply that Rademacher complexity of DNNs can be sufficiently controlled by the product of norms of weight matrices, especially the spectral norm and the Frobenius norm. In this section, we use $\|W\|_2$ to denote the spectral norm, which is the maximum singular value of W . And we use $\|W\|_{p,q}$ to denote $L_{p,q}$ -norm, defining as

$$\|W\|_{p,q} = \left(\sum_j \left(\sum_i |w_{ij}|^p \right)^{q/p} \right)^{1/q}.$$

When $p = q = 2$, this refers to Frobenius norm, denoted by $\|W\|_F$. The ℓ_p -norm of a vector \mathbf{x} is denoted by $\|\mathbf{x}\|_p$.

Reference [28] mainly focused on Relu and $L_{p,q}$ -norm of weight matrices. They extended prior work on per-unit regularization for two-layer networks to the overall norm regularization, which corresponds to the commonly used technique of weight decay.

Theorem 44.3.1 ([28]) *For any $L, q \geq 1$, any $1 \leq p < \infty$, and any n i.i.d. samples $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ from \mathcal{X} , we have*

$$\widehat{\mathfrak{R}}_n(\mathcal{F}_{\mathcal{W}_L}) \leq \sqrt{\frac{1}{n} \prod_{j=1}^L \|W_j\|_{p,q}^2 \left(2d^{\lfloor \frac{1}{p^*} - \frac{1}{q} \rfloor_+}\right)^{2(L-1)} \min\{p^*, 4 \log(2d_0)\} \max_i \|\mathbf{x}_i\|_{p^*}^2},$$

where p^* is such that $\frac{1}{p} + \frac{1}{p^*} = 1$.

When $p = q = 2$, and if the ℓ_2 -norm of each sample is bounded by B , the upper bound of $\widehat{\mathfrak{R}}_n(\mathcal{F}_{\mathcal{W}_L})$ can be reduced to

$$\mathcal{O}\left(\frac{B^{2L} \prod_{j=1}^L \|W_j\|_F}{\sqrt{n}}\right).$$

Although this bound has no explicit dependence on the network width, it has a very strong exponential dependence on the network depth L , even if $\|W_j\|_F \leq 1$ for all j .

Reference [6] shows the generalization error of DNN is proportional to its Lipschitz constant, particularly the Lipschitz constant normalized by the margin of the predictor. It is easy to prove that the Lipschitz constant of DNNs is actually the product of spectral norms of all weight matrices. A detailed description of the margin of the predictor is as follows. Given a neural network $f_{\mathcal{W}_L}$ defined in Eq. (44.7), mapping from \mathbb{R}^{d_0} to \mathbb{R}^{d_L} , where d_L is also the number of classes, the most natural way to predict classes is to select the output coordinate with the largest magnitude, meaning $\mathbf{x} \mapsto \arg \max_j f_{\mathcal{W}_L}(\mathbf{x})_j$. Then, the margin is defined as $f_{\mathcal{W}_L}(\mathbf{x})_y - \max_{j \neq y} f_{\mathcal{W}_L}(\mathbf{x})_j$, which measures the gap between the output for the correct label and other labels. The following theorem shows the generalization bound for DNNs under this setting.

Theorem 44.3.2 ([6]) *Let $(\sigma_1, \dots, \sigma_L)$ be 1-Lipschitz activation function satisfying $\sigma_i(0) = 0$. Then for $(\mathbf{x}, y), (\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)$ drawn i.i.d. from \mathcal{D} , with probability at least $1 - \eta$ over $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, every margin $\gamma > 0$ and network $f_{\mathcal{W}_L} : \mathbb{R}^{d_0} \mapsto \mathbb{R}^{d_L}$ satisfy*

$$\begin{aligned} & \Pr \left[\arg \max_j f_{\mathcal{W}_L}(\mathbf{x})_j \neq y \right] \\ & \leq \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left[f_{\mathcal{W}_L}(\mathbf{x}_i)_{y_i} \leq \gamma + \max_{j \neq y_i} f_{\mathcal{W}_L}(\mathbf{x}_i)_j \right] \\ & + \tilde{\mathcal{O}} \left(\frac{\|X\|_F R_{\mathcal{W}}}{\gamma n} \ln(d) + \sqrt{\frac{\ln(1/\eta)}{n}} \right), \end{aligned}$$

where $\|X\|_F = \sqrt{\sum_i \|\mathbf{x}_i\|_2^2}$ and

$$R_{\mathcal{W}} := \left(\prod_{j=1}^L \|W_j\|_2 \right) \left(\sum_{j=1}^L \frac{\|W_j^\top\|_{2,1}^{2/3}}{\|W_j\|_2^{2/3}} \right)^{3/2}.$$

As we can see, the above result mainly relies on the term $R_{\mathcal{W}}$, which actually polynomially depends on the depth of DNNs. Since for any weight matrix W_j ,

$$\frac{\sum_l \sqrt{\sum_k W_j(l, k)^2}}{\|W_j\|_2} \geq \frac{\|W_j\|_F}{\|W_j\|_2} \geq 1,$$

this means $R_{\mathcal{W}}$ can never be smaller than $\prod_{j=1}^L \|W_j\|_2 \sqrt{L^3}$.

Recently, [13] proposed a nearly size-independent sample complexity of DNNs based on Frobenius norms of weight matrices. For linear predictor $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$, we have proved that the Rademacher complexity is upper bounded by $\mathcal{O}(\|\mathbf{w}\|_2 \|\mathbf{x}\|_2 / \sqrt{n})$, which is apparently independent of the number of parameters. Motivated by this observation, they came up with a nearly size-independent bound under some mild conditions.

Theorem 44.3.3 ([13]) *Let $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ be n i.i.d. samples from $\mathcal{X} = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq B\}$. Let $(\sigma_1, \dots, \sigma_L)$ be 1-Lipschitz activation function satisfying $\sigma_i(0) = 0$. Assume that parameter matrices W_j satisfy $\prod_{j=1}^L \|W_j\|_2 \geq \Gamma$. With additional assumptions about the loss function ℓ and DNNs class $\mathcal{F}_{\mathcal{W}_L}$, it holds that*

$$\begin{aligned} \widehat{\mathfrak{R}}_n(\ell \circ \mathcal{F}) & \leq \mathcal{O} \left(\frac{B \prod_{j=1}^L \|W_j\|_F}{\gamma} \right. \\ & \left. \min \left\{ \frac{\log^{3/4}(n) \sqrt{\log \left(\frac{1}{\Gamma} \prod_{j=1}^L \|W_j\|_F \right)}}{n^{1/4}}, \sqrt{\frac{L}{n}} \right\} \right), \end{aligned}$$

where $\log(z) := \max\{1, \log(z)\}$ and γ is the margin.

Ignoring logarithmic factors and replacing the min by its first argument, the bound in is at most

$$\tilde{\mathcal{O}} \left(\frac{B \prod_{j=1}^L \|W_j\|_F}{\gamma} \sqrt{\frac{\log \left(\frac{1}{\Gamma} \prod_{j=1}^L \|W_j\|_F \right)}{\sqrt{n}}} \right).$$

Assuming that $\prod_{j=1}^L \|W\|_F$ and $\prod_{j=1}^L \|W\|_F / \Gamma$ are bounded by a constant, a bound that does not depend on the width or depth of the network can be obtained.

The above works discussed DNNs without including bias terms. However, [39] provided some examples to state that these conclusions may fail when bias terms are added for each hidden layer. For the bias-included DNNs, the dimension of each weight matrix W_j is now $\mathbb{R}^{d_j} \times \mathbb{R}^{d_{j-1}+1}$, and we denote such DNNs class as $\mathcal{F}_{\mathcal{W}_{L,b}}$. An $L_{p,q}$ -norm based upper bound on the Rademacher complexity of $\mathcal{F}_{\mathcal{W}_{L,b}}$ is established. For simplicity, they assume that the $L_{p,q}$ norms of weight matrices of hidden layers are the same, i.e., $\|W_j\|_{p,q} \equiv c$ for $j = 1, \dots, L-1$, and the norm of output layer is controlled by a constant, i.e., $\|W_L\|_{p,q} \leq c_o$.

Theorem 44.3.4 ([39]) For any $q \geq 1$, $c, c_o > 0$, and n i.i.d. samples $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ from \mathcal{X} , we have

1.
$$\widehat{\mathfrak{R}}_n(\mathcal{F}_{\mathcal{W}_{L,b}}) \leq \mathcal{O} \left(c_o c^{L-1} \sqrt{\frac{L-1}{n}} \right).$$

2. for $p \in (1, \infty)$,

$$\widehat{\mathfrak{R}}_n(\mathcal{F}_{\mathcal{W}_{L,b}}) \leq \mathcal{O} \left(c_o \sqrt{\frac{L-1}{n}} \left(\sum_{i=2}^{L-1} c^{L-i} \prod_{l=i}^{L-1} d_l^{\left[\frac{1}{p^*} - \frac{1}{q}\right]_+} \right) \right),$$

where p^* is such that $\frac{1}{p} + \frac{1}{p^*} = 1$.

In Theorem 44.3.4, it is easy to show that the result has no dependence on widths if $q \in [1, p^*]$.

As a summary, we can see that the Rademacher complexity of DNNs with and without bias terms can be controlled by the product of norms of weight matrices. This infers that the outstanding generalization performance of DNNs is not due to the large number of parameters but the norms of weight matrices. Under reasonable assumptions and appropriate norm constraints, DNNs can achieve both high training accuracy and testing accuracy even if we use DNNs with small width and depth. Typically in practice, we can add weight decay [20] or use spectral weight normalization [41] to improve the generalization behavior of DNNs.

44.4 Learning Theory for Robust Adversarial Learning

Many recent studies have shown that DNNs are actually vulnerable to adversarial attacks [7, 11, 36]. The adversarial inputs are called *adversarial examples*, which are typically generated by adding small perturbations that are imperceptible to human eyes [35]. Although many deep learning models achieve the state-of-the-art performance in benchmark datasets, they perform poorly on these adversarial examples. For example, the adversarial test accuracy on CIFAR10 is reported as only 47% in [24], instead the natural test accuracy on CIFAR10 is around 95% [32]. Hence, it is necessary to study the adversarial robust generalization property of DNNs to theoretically gain deeper understanding of this problem.

As in the previous sections, assume that (\mathbf{x}, y) is from an unknown distribution \mathcal{D} . For adversarial learning, we focus on the ℓ_∞ white box adversarial attacks where an adversary is allowed to observe the trained model. We choose some \mathbf{x}' such that $\|\mathbf{x}' - \mathbf{x}\|_\infty \leq \epsilon$ and $\ell(f(\mathbf{x}'), y)$ is maximized. Therefore, the adversarial robust model is learned by minimizing the *empirical adversarial risk*, that is,

$$\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \max_{\|\mathbf{x}'_i - \mathbf{x}_i\|_\infty \leq \epsilon} \ell(f(\mathbf{x}'_i), y_i), \quad (44.8)$$

where $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ are n i.i.d. training samples from \mathcal{D} . The generalization behavior of the adversarial risk when \mathcal{F} is a set of predictors is the difference between the *expected adversarial risk*

$$\mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}} \left[\max_{\|\mathbf{x}' - \mathbf{x}\|_\infty \leq \epsilon} \ell(f(\mathbf{x}'), y) \right]$$

and the empirical adversarial risk in (44.8).

Let us start from simple predictors. Reference [40] proved tight upper and lower bounds for the adversarial Rademacher complexity of binary classifiers. Assume $f_{\mathbf{w}}(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle$ is the predictor, where $\mathbf{w} \in \mathbb{R}^d$. The label is then predicted as the sign of $f_{\mathbf{w}}(\mathbf{x})$. Since the label space $\mathcal{Y} = \{-1, 1\}$, the loss $\ell(f(\mathbf{x}), y)$ can be written as $\ell(f(\mathbf{x}), y) = \phi(y \langle \mathbf{w}, \mathbf{x} \rangle)$, where $\phi(\cdot) : \mathbb{R} \mapsto [0, A]$ is monotonically nonincreasing and L_ϕ -Lipschitz. Under the adversarial setting, we therefore have

$$\max_{\|\mathbf{x}' - \mathbf{x}\|_\infty \leq \epsilon} \ell(f_{\mathbf{w}}(\mathbf{x}'), y) = \phi \left(\min_{\|\mathbf{x}' - \mathbf{x}\|_\infty \leq \epsilon} y \langle \mathbf{w}, \mathbf{x}' \rangle \right)$$

Then it is easy to derive the Rademacher complexities with adversarial attacks. The following theorem shows that the adversarial Rademacher complexity is always at least as large as the Rademacher complexity in the natural setting.

Theorem 44.4.1 ([40]) Let $\mathcal{F} = \{f_{\mathbf{w}}(\mathbf{x}) : \|\mathbf{w}\|_p \leq c\}$ and $\tilde{\mathcal{F}} = \{\min_{\|\mathbf{x}' - \mathbf{x}\|_\infty \leq \epsilon} y \langle \mathbf{w}, \mathbf{x}' \rangle : \|\mathbf{w}\|_p \leq c\}$. Suppose that $\frac{1}{p} + \frac{1}{q} = 1$. Then, there exists a universal constant $C \in (0, 1)$ such that

$$\max \left\{ \mathfrak{R}_n(\mathcal{F}), C\epsilon c \frac{d^{\frac{1}{q}}}{\sqrt{n}} \right\} \leq \mathfrak{R}_n(\tilde{\mathcal{F}}) \leq \mathfrak{R}_n(\mathcal{F}) + \epsilon c \frac{d^{\frac{1}{q}}}{\sqrt{n}}.$$

For neural network classifiers, deriving the adversarial Rademacher complexity is more sophisticated, because Eq. (44.8) is an intractable optimization problem. The well-known fast gradient sign method (FGSM) [15] proposes to compute the adversarial examples as

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f(\mathbf{x}_i), y_i)), \quad i = 1, \dots, n. \quad (44.9)$$

The class of DNNs with spectral norm constraints and under such adversarial attacks is defined as

$$\tilde{\mathcal{F}}_{\mathcal{W}_L} = \left\{ \mathbf{x} \mapsto f_{\mathcal{W}_L}(\tilde{\mathbf{x}}) : \tilde{\mathbf{x}} = \mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)); \right. \\ \left. \|W_j\|_2 \leq c_j; \text{rank}(W_j) \leq r_j, \forall j \right\}, \quad (44.10)$$

where $f_{\mathcal{W}_L}(\mathbf{x})$ is defined in Eq. (44.7) and c_j and r_j are positive constants. The Rademacher complexity of $\tilde{\mathcal{F}}_{\mathcal{W}_L}$ is given as follows.

Theorem 44.4.2 Given $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ from $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^{d_0} : \|\mathbf{x}\| \leq B\}$, each \mathbf{x}_i is perturbed by $\epsilon \cdot \text{sign}(\nabla_{\mathbf{x}_i} \ell(f_{\mathcal{W}_L}(\mathbf{x}_i), y_i))$. Assume that the activation function $\sigma(\cdot)$ is 1-Lipschitz and 1-smooth. Assume that the loss function $\ell(\cdot, y)$ is 1-Lipschitz and 1-smooth for any fixed label y , and $\min_{p \in [d]} |\nabla_{\mathbf{x}_i}^{(p)} \ell(f_{\mathcal{W}_L}(\mathbf{x}_i), y_i)| \geq \kappa$ holds for a constant $\kappa > 0$, where $\nabla_{\mathbf{x}_i}^{(p)} \ell(f_{\mathcal{W}_L}(\mathbf{x}_i), y_i)$ is the p -th element of $\nabla_{\mathbf{x}_i} g(f_{\mathcal{W}_L}(\mathbf{x}_i), y_i)$, for $p = 1, \dots, d$. Then $\mathfrak{R}_n(\ell \circ \tilde{\mathcal{F}}_{\mathcal{W}_L})$ is upper bounded by

$$\mathcal{O} \left(\frac{\prod_{j=1}^L c_j (B + \sqrt{d}\epsilon)}{\sqrt{n}} \sqrt{d \sum_{j=1}^L r_j \ln \left(L\sqrt{n} \left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon} \right) \right)} \right),$$

where $\Theta = \epsilon \frac{1}{\kappa} (1 + \frac{1}{\kappa} \prod_{j=1}^L c_j) \prod_{j=1}^L c_j (1 + \frac{B}{L} \sum_{j=1}^L (j \prod_{k=1}^j c_k))$

This theorem shows that the effect of adversarial attacks on generalization performance is an additional linear term with ϵ . Besides an extra logarithm term, the coefficient of this linear term for ϵ includes the Lipschitz constant of the neural network, the width of the neural network, and the sum of ranks of weight matrices. Assuming the ranks are all equal

to r , $\sum_{j=1}^L r_j$ turns to be a linear function of the depth, which implies that the depth influences the generalization error of adversarial learning to a certain extent.

Several interesting observations will lead this upper bound to a tighter one. An L -layer neural network $f_{\mathcal{W}_L} : \mathbb{R}^{d_0} \rightarrow \mathbb{R}^{d_L}$ is a $\prod_{j=1}^L \|W_j\|_2$ -Lipschitz continuous function if all activation functions are 1-Lipschitz. It is easy to show that the covering number of DNNs class is independent of the network depth if we regard the whole network as a Lipschitz continuous function. In addition, weight matrices tend to be low rank in many empirical results, and dropout can be treated as the low-rank regularization. These take-home points motivate us to decompose $f_{\mathcal{W}_L}$ as a shallow network and a low-dimensional Lipschitz continuous function. Relying on this decomposition, we established a tighter upper bound on the Rademacher complexity in adversarial setting and further limit the effect of adversarial perturbations on the adversarial generalization performance.

In particular, suppose $W_l = U_l \Sigma_l V_l^\top$ for $\forall l \in \{1, \dots, L\}$, where U_l and V_l are column-orthogonal matrices, and $\Sigma_l \in \mathbb{R}^{r_l \times r_l}$ is a diagonal matrix whose entries are nonzero singular values of W_l . Then we rewrite $f_{\mathcal{W}_L}(\tilde{\mathbf{x}})$ as

$$f_{\mathcal{W}_L}(\tilde{\mathbf{x}}) = h_{r_l} \circ f_{\mathcal{W}_l}(\tilde{\mathbf{x}}),$$

where

$$f_{\mathcal{W}_l}(\tilde{\mathbf{x}}) = V_l^\top \sigma_{l-1}(\dots \sigma_1(W_l(\tilde{\mathbf{x}})))$$

is a depth- l neural network and

$$h_{r_l}(\mathbf{z}) = \sigma_L(W_L \sigma_{L-1}(W_{L-1} \dots \sigma_l(U_l \Sigma_l \mathbf{z})))$$

is a Lipschitz continuous function with low-dimensional input, mapping from \mathbb{R}^{r_l} to \mathbb{R}^{d_L} . The composition implies that $\ell \circ \tilde{\mathcal{F}}_{\mathcal{W}_L}$ is a subset of $\ell \circ \mathcal{H}_{r_l} \circ \tilde{\mathcal{F}}_{\mathcal{W}_l}$, where

$$\mathcal{H}_{r_l} = \left\{ h_{r_l}(\mathbf{z}) \mid \|\mathbf{z}\| \leq \prod_{j=1}^{l-1} \|W_j\|_2 (B + \sqrt{d}\epsilon), \right. \\ \left. \text{Lipschitz constant is } \prod_{j=l}^L \|W_j\|_2 \right\}. \quad (44.11)$$

According to the properties of Rademacher complexity, we have

$$\widehat{\mathfrak{R}}_n(\ell \circ \tilde{\mathcal{F}}_{\mathcal{W}_L}) \leq \widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \tilde{\mathcal{F}}_{\mathcal{W}_l}).$$

Since this decomposition holds true for any l , we further obtain the upper bound on $\widehat{\mathfrak{R}}_n(\ell \circ \tilde{\mathcal{F}}_{\mathcal{W}_L})$ by choosing the minimum among all $\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \tilde{\mathcal{F}}_{\mathcal{W}_l})$ for $l = 1, \dots, L$.

Lemma 44.4.1 Under the same assumptions in Theorem 44.4.2, and define $\Lambda = \prod_{j=1}^L c_j (B + \sqrt{d}\epsilon)$, we have

1. When $r_l = 1$, $\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \widetilde{\mathcal{F}}_{\mathcal{W}_l})$ satisfies

$$R_l^1 := \mathcal{O}\left(\frac{\Lambda}{\sqrt{n}} \sqrt{d \sum_{j=1}^l r_j \ln\left(l\sqrt{n}\left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon}\right)\right)}\right).$$

2. When $r_l = 2$, $\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \widetilde{\mathcal{F}}_{\mathcal{W}_l})$ satisfies

$$R_l^2 := \mathcal{O}\left(\frac{\Lambda}{\sqrt{n}} \left(\sqrt{(\ln \sqrt{n})^3} + \sqrt{d \sum_{j=1}^l r_j \ln\left(l\sqrt{n}\left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon}\right)\right)}\right)\right).$$

3. When $r_l \geq 3$, $\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \widetilde{\mathcal{F}}_{\mathcal{W}_l})$ satisfies

$$R_l^{r_l} := \mathcal{O}\left(\frac{\Lambda}{\sqrt[n]{n}} \left(\frac{\sqrt{24^{r_l} \ln(\sqrt[n]{n})}}{r_l} + \sqrt{d \sum_{j=1}^l r_j \ln\left(l\sqrt[n]{n}\left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon}\right)\right)}\right)\right).$$

Theorem 44.4.3 For $i = 1, \dots, d$, define

$$l^{(i)} = \min_{j \in [L]} \{j : \text{rank}(W_j) = i\}.$$

Then,

$$\widehat{\mathfrak{R}}_n(g \circ \widetilde{\mathcal{F}}_{\mathcal{W}_L}) \leq \min_{i \in [d]} R_{l^{(i)}}^i,$$

where the $R_{l^{(i)}}^i$ is defined in Lemma 44.4.1.

We ignore logarithmic factors for simplicity. Consider two different scenarios. The first case is that there are low rank matrices with rank 1 or 2 at layer l among weight matrices. Theorem 44.4.3 shows that the upper bound on the Rademacher complexity of $\ell \circ \widetilde{\mathcal{F}}_{\mathcal{W}_L}$ is at most of order

$$\frac{\left(\prod_{j=1}^L \|W_j\|_2\right) (B + \sqrt{d}\epsilon) \sqrt{d \sum_{j=1}^l r_j}}{\sqrt{n}}.$$

The second case is that the ranks of weight matrices are all greater than or equal to 3. Theorem 44.4.3 shows that the upper bound is at most of order

$$\left(\prod_{j=1}^L \|W_j\|_2\right) (B + \sqrt{d}\epsilon) \cdot \min\left(\sqrt{\frac{d \sum_{j=1}^L r_j}{n}}, \frac{\sqrt{d \sum_{j=1}^l r_j} + \sqrt{24^{r_l}/r_l}}{\sqrt[n]{n}}\right).$$

These bounds are depth-free, depend on the Lipschitz constant $\prod_{j=1}^L \|W_j\|_2$, and have a linear relationship with ϵ whose coefficient is linear in the width d .

Following the above conclusion, it is clear that the Rademacher complexity of DNNs under adversarial setting only relies on the shallow part if a low rank weight matrix exists. Correspondingly, the linear coefficient of adversarial perturbations reduces to $\mathcal{O}(\sqrt{\sum_{j=1}^l r_j})$. Thus, adversarial attacks have a smaller influence on the Rademacher complexity and also the generalization behavior of the DNNs. To improve the training accuracy and testing accuracy for experiments, we may adopt spectral normalization and dropout.

44.5 Example: Generalization Bounds for Logistic Regression

In this section, we consider a widely used model in statistical machine learning, which is the logistic regression, to illustrate the previous theoretical results. Let (\mathbf{x}, y) be the sample from $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^{d_0} : \|\mathbf{x}\|_2 \leq B\}$ and $\mathcal{Y} = \{0, 1\}$ are input domain and output domain, respectively. The non-linear logistic regression model is

$$f(\mathbf{x}) := \mathbb{P}(Y = 1 | \mathbf{x}) = \frac{\exp(h(\mathbf{x}))}{1 + \exp(h(\mathbf{x}))},$$

where, for simplicity, $h(\mathbf{x}) = \mathbf{w}^T \sigma(W\mathbf{x})$ is modeled by a single hidden layer neural network, σ is the Sigmoid activation function, and $W \in \mathbb{R}^{d_1 \times d_0}$ and $\mathbf{w} \in \mathbb{R}^{d_1}$ are weight parameters. Let \mathcal{F} be a class of logistic regressions functions with spectral normalization,

$$\mathcal{F} = \{\mathbf{x} \mapsto f(\mathbf{x}) : \|W\|_2 \leq c_1, \|\mathbf{w}\|_2 \leq c_2, \text{rank}(W) \leq r_1\}.$$

To find the optimal estimations for parameters, we minimize the negative log-likelihood

$$\ell(f(\mathbf{x}), y) = -y \ln(f(\mathbf{x})) - (1 - y) \ln(1 - f(\mathbf{x})).$$

It is easy to show that the Sigmoid activation function is 1-Lipschitz and 1-smooth. The log loss also satisfies these conditions if the input is lower bounded by 1. Furthermore, we may set $\kappa \leq c \min_p \{|\mathbf{w}^T W^{(p)}|\}$ for some constant c , where $W^{(p)}$ is the p -th column of W for $p = 1, \dots, d_0$. This satisfies the condition for κ in Theorem 44.4.2. Given n i.i.d. samples $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)$ from $\mathcal{X} \times \mathcal{Y}$, we consider the FGSM attack, that is, the perturbed predictors are $\tilde{\mathbf{x}}_i = \mathbf{x}_i + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f(\mathbf{x}_i), y_i))$ for $i = 1, \dots, n$. Then, with probability at least $1 - \eta$ for $\eta \in (0, 1)$, the generalization error under the FGSM attack satisfies

$$\begin{aligned} & \mathbb{E}_{(x,y)} \ell(f(\tilde{x}), y) - \frac{1}{n} \sum_{i=1}^n \ell(f(\tilde{x}_i), y_i) \\ & \leq \mathcal{O} \left(\frac{c_1 c_2 (B + \sqrt{d}\epsilon)}{\sqrt{n}} \right. \\ & \quad \left. \sqrt{d(1+r_1) \ln \left(\sqrt{n} \left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon} \right) \right)} + \sqrt{\frac{\ln(1/\eta)}{n}} \right), \end{aligned}$$

where $d = \max\{d_0, d_1\}$ and $\Theta = \epsilon \frac{c_1 c_2}{\kappa} (1 + \frac{c_1 c_2}{\kappa}) (1 + \frac{B(c_1 + 2c_1 c_2)}{2})$.

The upper bound on generalization error in natural learning case is straightforward by setting $\epsilon = 0$. That is,

$$\begin{aligned} & \mathbb{E}_{(x,y)} \ell(f(x), y) - \frac{1}{n} \sum_{i=1}^n \ell(f(x_i), y_i) \leq \mathcal{O} \\ & \left(\frac{c_1 c_2 B \sqrt{d(1+r_1) \ln(\sqrt{n})}}{\sqrt{n}} + \sqrt{\frac{\ln(1/\eta)}{n}} \right). \end{aligned}$$

The crucial observation is that, for logistic regression, the adversarial generalization bound is always larger than its natural learning counterpart, but the effect of adversarial perturbations can be limited under our weight normalization framework.

44.6 Discussion

We have presented a systematic framework of statistical learning for supervised learning problems. This includes common technical tools, generalization error bounds for DNNs, and the sample complexity of DNNs for robust adversarial learning with ℓ_∞ attacks. We have omitted several important theoretical results for other deep learning models, which include generative adversarial networks (GANs) [1, 14], deep graphical models [19], deep reinforcement learning [23, 25], etc. In the end, we point out a few future directions and hope that these open problems serve as a stimulus for further statistical learning research.

- *Generative Models.* There are two main approaches for generative modeling. The first one is called *variational auto-encoders* (VAEs) [18], which use variational inference to learn a model by maximizing the lower bound of the likelihood function. VAEs have elegant theoretical foundations, but the drawback is that they tend to produce blurry images. The second approach is called *generative adversarial networks* (GANs) [14], which learn a model by using a powerful discriminator to distinguish between real data points and generative data points. GANs produce

more visually realistic images but suffer from the unstable training and the mode collapse problem. How to provide a unifying framework combining the best of VAEs and GANs in a principled way is yet to be discovered.

- *Trade-off between robustness and standard accuracy.* The trade-off between robustness and standard accuracy has been consistently reported in the machine learning literature. Although the problem has been widely studied to understand and explain this trade-off, no studies have shown the possibility of a no trade-off solution. Is there any way to overcome this trade-off?
- *Latent variable models and the MLE.* Latent variable models are powerful and cover a broad range of statistical and machine learning models, such as Bayesian models, linear mixed models, and Gaussian mixture models. Existing methods often suffer from two major challenges in practice: (a) A proper latent variable distribution is difficult to be specified, and (b) making an exact likelihood inference is formidable due to the intractable computation. Is there any framework which can overcome these limitations and allows efficient and exact maximum likelihood inference?

Appendix

Proof of Theorem 44.4.2

To prove Theorem 44.4.2, we need the covering number of $\tilde{\mathcal{F}}_{\mathcal{W}_L}$ first.

Lemma 44.6.1 *Under the same conditions as Theorem 44.4.2, we have the covering number of $\tilde{\mathcal{F}}_{\mathcal{W}_L}$ with respect to the metric ρ_n satisfies*

$$\mathcal{N}(\tau, \tilde{\mathcal{F}}_{\mathcal{W}_L}, \rho_n) \leq \left(\frac{9L(B + \sqrt{d}\epsilon + \Theta) \prod_{j=1}^L c_j}{\tau} \right)^{(2d+1) \sum_{j=1}^L r_j},$$

where ρ_n is the $L_2(P_n)$ metric and $\Theta = \epsilon \frac{1}{\kappa} (1 + \frac{1}{\kappa} \prod_{j=1}^L c_j) \prod_{j=1}^L c_j (1 + \frac{B}{L} \sum_{j=1}^L (j \prod_{k=1}^j c_k))$.

We use the following Lemma 44.6.2, Lemma 44.6.3 and Lemma 44.6.4 to prove Lemma 44.6.1.

Lemma 44.6.2 *Let $S_{c,r} = \{W : W \in \mathbb{R}^{d_2 \times d_1}, \|W\|_2 \leq c, \text{rank}(W) \leq r\}$. Then there exists an τ -covering of $S_{c,r}$ with respect to the spectral norm obeying*

$$\mathcal{N}(\tau, S_{c,r}, \|\cdot\|_2) \leq \left(\frac{9c}{\tau} \right)^{r(d_2+d_1+1)}.$$

Proof. We prove the lemma by extending the arguments from [9]. We do SVD of W in $S_{c,r}$,

$$W = U\Sigma V^\top = cU\frac{\Sigma}{c}V^\top := cU\tilde{\Sigma}V^\top, \quad (44.12)$$

where $\Sigma \in \mathbb{R}^{r \times r}$ is the diagonal matrix with singular values and $U \in \mathbb{R}^{d_2 \times r}$ and $V \in \mathbb{R}^{d_1 \times r}$ are column orthogonal matrices. Thus, $\|U\|_2 = \|V\|_2 = 1$, and $\|\tilde{\Sigma}\|_2 \leq 1$. We will construct a τ -covering for $S_{c,r}$ by covering the set of U , $\tilde{\Sigma}$ and V . we assume $d_1 = d_2 = d$ for simplicity.

Let Γ be the set of diagonal matrices with nonnegative entries and spectral norm less than 1. We take Γ' to be an $\tau/(3c)$ -net for Γ with

$$|\Gamma'| \leq \left(\frac{9c}{\tau}\right)^r.$$

Let $O_{d,r} = \{U \in \mathbb{R}^{d \times r} : \|U\|_2 = 1\}$. There also exists an $\tau/(3c)$ -net $O'_{d,r}$ for $O_{d,r}$ obeying

$$|O'_{d,r}| \leq \left(\frac{9c}{\tau}\right)^{dr}.$$

We now let $S'_{c,r} = \{cU'\Sigma'V'^\top : U', V' \in O'_{d,r}, \Sigma' \in \Gamma'\}$. Thus,

$$|S'_r| \leq |O'_{d,r}|^2 |\Gamma'| \leq \left(\frac{9c}{\tau}\right)^{r(2d+1)}.$$

It remains to show that there exists $S'_{c,r}$ for $S_{c,r}$, such that $\|W - W'\|_2 \leq \tau$.

$$\begin{aligned} & \|W - W'\|_2 \\ &= c \left\| U\tilde{\Sigma}V^\top - U'\Sigma'V'^\top \right\|_2 \\ &= c \left\| U\tilde{\Sigma}V^\top - U'\tilde{\Sigma}V^\top + U'\tilde{\Sigma}V^\top - U'\Sigma'V'^\top \right. \\ & \quad \left. + U'\Sigma'V'^\top - U'\Sigma'V'^\top \right\|_2 \\ &\leq c \left(\left\| (U - U')\tilde{\Sigma}V^\top \right\|_2 + \left\| U'(\tilde{\Sigma} - \Sigma')V^\top \right\|_2 \right. \\ & \quad \left. + \left\| U'\Sigma'(V - V')^\top \right\|_2 \right) \end{aligned}$$

For the first term,

$$\left\| (U - U')\tilde{\Sigma}V^\top \right\|_2 \leq \|U - U'\|_2 \|\tilde{\Sigma}\|_2 \|V\|_2 \leq \frac{\tau}{3c}.$$

The same argument gives $\|U'\Sigma'(V - V')^\top\|_2 \leq \tau/(3c)$. For the second term,

$$\left\| U'(\tilde{\Sigma} - \Sigma')V^\top \right\|_2 \leq \|\tilde{\Sigma} - \Sigma'\|_2 \leq \frac{\tau}{3c}.$$

Therefore, $\|W - W'\|_2 \leq \tau$. This completes the proof. \square

Lemma 44.6.3 *Under the same conditions as Theorem 44.4.2, given $\mathcal{W}_L = (W_1, \dots, W_L)$, there exists $\mathcal{W}'_L = (W'_1, \dots, W'_L)$, where W'_j is from the τ_j -covering of S_{c_j, r_j} , for $j = 1, \dots, L$. Then*

$$\begin{aligned} & \left\| \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y) \right\|_2 \leq \prod_{j=1}^L c_j \sum_{j=1}^L \\ & \left(\frac{\tau_j}{c_j} + B \prod_{k=1}^j c_k \sum_{k=1}^j \frac{\tau_k}{c_k} \right). \end{aligned}$$

Proof. According to the Lipschitz and smooth assumptions of the activation function and loss function, we have

$$\begin{aligned} & \left\| \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y) \right\|_2 \\ & \leq \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_L}(\mathbf{x}) (\nabla \ell)(f_{\mathcal{W}_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) (\nabla \ell)(f_{\mathcal{W}_L}(\mathbf{x}), y) \right\|_2 \\ & \quad + \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) (\nabla \ell)(f_{\mathcal{W}_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) (\nabla \ell)(f_{\mathcal{W}'_L}(\mathbf{x}), y) \right\|_2 \\ & \leq \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_L}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) \right\|_2 + \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) \right\|_2 \\ & \quad \left\| (\nabla \ell)(f_{\mathcal{W}_L}(\mathbf{x}), y) - (\nabla \ell)(f_{\mathcal{W}'_L}(\mathbf{x}), y) \right\|_2 \\ & \leq \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_L}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) \right\|_2 + \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) \right\|_2 \\ & \quad \left\| f_{\mathcal{W}_L}(\mathbf{x}) - f_{\mathcal{W}'_L}(\mathbf{x}) \right\|_2 \end{aligned} \quad (44.13)$$

$$\leq \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_L}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) \right\|_2 + \prod_{j=1}^L c_j^2 B \sum_{j=1}^L \frac{\tau_j}{c_j}. \quad (44.14)$$

Note that (44.13) holds because $\ell(\cdot)$ is 1-smooth, and $\|f_{\mathcal{W}_L}(\mathbf{x}) - f_{\mathcal{W}'_L}(\mathbf{x})\|_2 \leq \prod_{j=1}^L c_j B \sum_{j=1}^L \tau_j / c_j$ in (44.14) can be obtained from [6]. Next, we prove the following inequality by induction.

$$\begin{aligned} & \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_L}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) \right\|_2 \leq \prod_{j=1}^L c_j \sum_{j=1}^L \\ & \left(\frac{\tau_j}{c_j} + B \prod_{k=1}^{j-1} c_k \sum_{k=1}^{j-1} \frac{\tau_k}{c_k} \right). \end{aligned} \quad (44.15)$$

When $L = 0$, $f_{\mathcal{W}_0}(\mathbf{x}) = \mathbf{x}$, $\left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_0}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_0}(\mathbf{x}) \right\|_2 = 0$. Assume (44.15) holds when there are $L - 1$ layers for DNN. Then, we have

$$\begin{aligned} & \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_L}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x}) \right\|_2 \\ &= \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_{L-1}}(\mathbf{x}) (\nabla \sigma_{L-1})(f_{\mathcal{W}_{L-1}}(\mathbf{x})) W_L^\top \right. \\ & \quad \left. - \nabla_{\mathbf{x}} f_{\mathcal{W}'_{L-1}}(\mathbf{x}) (\nabla \sigma_{L-1})(f_{\mathcal{W}'_{L-1}}(\mathbf{x})) W_L^\top \right\|_2 \\ &\leq \left\| \nabla_{\mathbf{x}} f_{\mathcal{W}_{L-1}}(\mathbf{x}) (\nabla \sigma_{L-1})(f_{\mathcal{W}_{L-1}}(\mathbf{x})) W_L^\top \right. \\ & \quad \left. - \nabla_{\mathbf{x}} f_{\mathcal{W}_{L-1}}(\mathbf{x}) (\nabla \sigma_{L-1})(f_{\mathcal{W}'_{L-1}}(\mathbf{x})) W_L^\top \right\|_2 \end{aligned}$$

$$\begin{aligned}
& + \|\nabla_{\mathbf{x}} f_{\mathcal{W}_{L-1}}(\mathbf{x})(\nabla \sigma_{L-1})(f_{\mathcal{W}'_{L-1}}(\mathbf{x}))W_L^\top \\
& \quad - \nabla_{\mathbf{x}} f_{\mathcal{W}_{L-1}}(\mathbf{x})(\nabla \sigma_{L-1})(f_{\mathcal{W}'_{L-1}}(\mathbf{x}))W_L^{\top T}\|_2 \\
& + \|\nabla_{\mathbf{x}} f_{\mathcal{W}_{L-1}}(\mathbf{x})(\nabla \sigma_{L-1})(f_{\mathcal{W}'_{L-1}}(\mathbf{x}))W_L^\top - \nabla_{\mathbf{x}} f_{\mathcal{W}'_{L-1}}(\mathbf{x}) \\
& \quad (\nabla \sigma_{L-1})(f_{\mathcal{W}'_{L-1}}(\mathbf{x}))W_L^{\top T}\|_2 \\
& \leq \prod_{j=1}^{L-1} c_j \|f_{\mathcal{W}_{L-1}}(\mathbf{x}) - f_{\mathcal{W}'_{L-1}}(\mathbf{x})\|_2 c_L \\
& \quad + \prod_{j=1}^{L-1} c_j \tau_j + \|\nabla_{\mathbf{x}} f_{\mathcal{W}_{L-1}}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\mathcal{W}'_{L-1}}(\mathbf{x})\|_2 c_L \\
& \leq \prod_{j=1}^L c_j \prod_{j=1}^{L-1} c_j B \sum_{j=1}^{L-1} \frac{\tau_j}{c_j} + \prod_{j=1}^{L-1} c_j \tau_j + \prod_{j=1}^L c_j \sum_{j=1}^{L-1} \\
& \quad \left(\frac{\tau_j}{c_j} + B \prod_{k=1}^{j-1} c_k \sum_{k=1}^{j-1} \frac{\tau_k}{c_k} \right) \\
& \leq \prod_{j=1}^L c_j \sum_{j=1}^L \left(\frac{\tau_j}{c_j} + B \prod_{k=1}^{j-1} c_k \sum_{k=1}^{j-1} \frac{\tau_k}{c_k} \right)
\end{aligned}$$

Combining (44.14) and (44.15), we can get the conclusion. \square

Lemma 44.6.4 *Under the same assumptions as Theorem 44.4.2, we have*

$$\begin{aligned}
& \|\text{sign}(\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)) - \text{sign}(\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y))\|_2 \\
& \leq \frac{1}{\kappa} \left(1 + \frac{1}{\kappa} \prod_{j=1}^L c_j \right) \prod_{j=1}^L c_j \sum_{j=1}^L \left(\frac{\tau_j}{c_j} + B \prod_{k=1}^j c_k \sum_{k=1}^j \frac{\tau_k}{c_k} \right)
\end{aligned}$$

Proof. According to the definition,

$$\text{sign}(\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)) = \Phi^{-1} \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y),$$

where

$$\Phi = \begin{pmatrix} |\nabla_{\mathbf{x}}^{(1)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)| & & & \\ & |\nabla_{\mathbf{x}}^{(2)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)| & & \\ & & \ddots & \\ & & & |\nabla_{\mathbf{x}}^{(d)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)| \end{pmatrix},$$

in which $\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)$ is the p -th element of $\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)$. Similarly, define

$$\Phi' = \text{diag}(|\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)|).$$

Then, for fixed y ,

$$\begin{aligned}
& \|\text{sign}(\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)) - \text{sign}(\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y))\|_2 \\
& = \|\Phi^{-1} \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y) - (\Phi')^{-1} \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2 \\
& \leq \|\Phi^{-1} \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y) - \Phi^{-1} \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2 \\
& \quad + \|\Phi^{-1} \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y) + (\Phi')^{-1} \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2 \\
& \leq \|\Phi^{-1}\|_2 \|\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2 \\
& \quad + \|\Phi^{-1} - (\Phi')^{-1}\|_2 \|\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2 \\
& \leq \left(\|\Phi^{-1}\|_2 + \frac{1}{\kappa^2} \|\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2 \right) \\
& \quad \|\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2 \tag{44.16}
\end{aligned}$$

$$\begin{aligned}
& \leq \frac{1}{\kappa} \left(1 + \frac{1}{\kappa} \prod_{j=1}^L c_j \right) \|\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2 \\
& \tag{44.17}
\end{aligned}$$

$$\begin{aligned}
& \leq \frac{1}{\kappa} \left(1 + \frac{1}{\kappa} \prod_{j=1}^L c_j \right) \prod_{j=1}^L c_j \sum_{j=1}^L \left(\frac{\tau_j}{c_j} + B \prod_{k=1}^j c_k \sum_{k=1}^j \frac{\tau_k}{c_k} \right). \\
& \tag{44.18}
\end{aligned}$$

Under the assumptions, (44.16) is obtained by

$$\begin{aligned}
\|\Phi^{-1} - (\Phi')^{-1}\|_2 & = \max_{p \in [d]} \\
& \frac{|\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)| - |\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)|}{|\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)| |\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)|} \\
& \leq \frac{1}{\kappa^2} \max_{p \in [d]} \left| |\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)| - |\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)| \right| \\
& \leq \frac{1}{\kappa^2} \max_{p \in [d]} |\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)| \\
& \leq \frac{1}{\kappa^2} \|\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y) - \nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y)\|_2.
\end{aligned}$$

Equation (44.17) is obtained by $\|\Phi^{-1}\|_2 = \max_{p \in [d]} (1/|\nabla_{\mathbf{x}}^{(p)} \ell(f_{\mathcal{W}_L}(\mathbf{x}), y)|) \leq 1/\kappa$ and

$$\nabla_{\mathbf{x}} \ell(f_{\mathcal{W}'_L}(\mathbf{x}), y) = \nabla_{\mathbf{x}} f_{\mathcal{W}'_L}(\mathbf{x})(\nabla \ell)(f_{\mathcal{W}'_L}(\mathbf{x}), y) \leq \prod_{j=1}^L c_j.$$

(44.18) is obtained from Lemma 44.6.3. \square

Now, we prove that under the assumptions of Lemma 44.6.3, given $\tau > 0$,

$$\rho_n = \sqrt{\left(\frac{1}{n} \sum_{i=1}^n \|f_{W_L}(\mathbf{x}_i + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}_i} \ell(f_{W_L}(\mathbf{x}_i), y))) - f_{W'_L}(\mathbf{x}_i + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}_i} \ell(f_{W'_L}(\mathbf{x}_i), y)))\|_2^2 \right)} \leq \tau$$

by choosing

$$\tau_j = \frac{\tau c_j}{L(B + \sqrt{d}\epsilon + \Theta) \prod_{i=1}^L c_i}. \quad (44.19)$$

First, we inductively prove that

$$\begin{aligned} & \|f_{W_L}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y))) \\ & - f_{W'_L}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W'_L}(\mathbf{x}), y)))\|_2 \\ & \leq \prod_{j=1}^L c_j \sum_{j=1}^L \frac{\tau_j}{c_j} (B + \sqrt{d}\epsilon) + \prod_{j=1}^L c_j \epsilon \\ & \quad \|\text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y)) - \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W'_L}(\mathbf{x}), y))\|_2 \end{aligned}$$

When $L = 0$, it is obvious that the above inequality holds.

Then,

$$\begin{aligned} & \|f_{W_L}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y))) \\ & - f_{W'_L}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W'_L}(\mathbf{x}), y)))\|_2 \\ & \leq \|W_L \sigma_{L-1}(f_{W_{L-1}}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y)))) \\ & \quad - W'_L \sigma_{L-1}(f_{W_{L-1}}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y))))\|_2 \\ & + \|W'_L \sigma_{L-1}(f_{W_{L-1}}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y)))) \\ & \quad - W'_L \sigma_{L-1}(f_{W'_{L-1}}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W'_L}(\mathbf{x}), y))))\|_2 \\ & \leq \tau_L \prod_{j=1}^{L-1} c_j (B + \sqrt{d}\epsilon) + c_L \|f_{W_{L-1}}(\mathbf{x} + \epsilon \cdot \text{sign} \\ & \quad (\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y))) - f_{W'_{L-1}}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W'_L}(\mathbf{x}), y)))\|_2 \\ & \leq \tau_L \prod_{j=1}^{L-1} c_j (B + \sqrt{d}\epsilon) + \prod_{j=1}^L c_j \sum_{j=1}^{L-1} \frac{\tau_j}{c_j} (B + \sqrt{d}\epsilon) \\ & \quad + \prod_{j=1}^L c_j \epsilon \|\text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y)) - \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W'_L}(\mathbf{x}), y))\|_2 \\ & \leq \prod_{j=1}^L c_j \sum_{j=1}^L \frac{\tau_j}{c_j} (B + \sqrt{d}\epsilon) + \prod_{j=1}^L c_j \epsilon \|\text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y)) \\ & \quad - \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W'_L}(\mathbf{x}), y))\|_2. \end{aligned}$$

Applying Lemma 44.6.4 and (44.19), we have

$$\begin{aligned} & \|f_{W_L}(\mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W_L}(\mathbf{x}), y))) - f_{W'_L}(\mathbf{x} + \epsilon \cdot \\ & \quad \text{sign}(\nabla_{\mathbf{x}} \ell(f_{W'_L}(\mathbf{x}), y)))\|_2 \end{aligned}$$

$$\begin{aligned} & \leq \prod_{j=1}^L c_j \left(\sum_{j=1}^L \frac{\tau_j}{c_j} (B + \sqrt{d}\epsilon) + \epsilon \frac{1}{\kappa} \left(1 + \frac{1}{\kappa} \prod_{j=1}^L c_j \right) \right. \\ & \quad \left. \prod_{j=1}^L c_j \sum_{j=1}^L \left(\frac{\tau_j}{c_j} + B \prod_{k=1}^j c_k \sum_{k=1}^j \frac{\tau_k}{c_k} \right) \right) \\ & \leq \tau \end{aligned}$$

Therefore, the covering number of $\tilde{\mathcal{F}}_{W_L}$ is

$$\begin{aligned} \mathcal{N}(\tau, \tilde{\mathcal{F}}_{W_L}, \rho_n) & \leq \prod_{j=1}^L \sup_{\substack{W_1, \dots, W_{j-1} \\ \forall i < j, W_i \in \mathcal{S}_{c_i, r_i}}} \\ & \quad \mathcal{N}\left(\tau_j, \left\{W_j : W_j \in \mathcal{S}_{c_j, r_j}\right\}, \|\cdot\|_2\right) \\ & \leq \prod_{j=1}^L \left(\frac{9c_j}{\tau_j}\right)^{r_j(d_j + d_{j-1} + 1)} \\ & \leq \left(\frac{9L(B + \sqrt{d}\epsilon + \Theta) \prod_{j=1}^L c_j}{\tau}\right)^{(2d+1) \sum r_j}. \end{aligned}$$

This completes the proof of the covering number.

By standard Dudley's entropy integral, we have

$$\begin{aligned} \hat{\mathfrak{R}}_n(\tilde{\mathcal{F}}_{W_L}) & \lesssim \inf_{\beta > 0} \left\{ \beta + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{\ln \mathcal{N}(\tau, \tilde{\mathcal{F}}_{W_L}, \rho_n) d\tau} \right\} \\ & \leq \inf_{\beta > 0} \left\{ \beta + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \right. \\ & \quad \left. \sqrt{(2d+1) \sum_{j=1}^L r_j \ln \left(\frac{9L(B + \sqrt{d}\epsilon + \Theta) \prod_{j=1}^L c_j}{\tau} \right)} d\tau \right\} \\ & \leq \inf_{\beta > 0} \left\{ \beta + \frac{\alpha}{\sqrt{n}} \right. \\ & \quad \left. \sqrt{(2d+1) \sum_{j=1}^L r_j \ln \left(\frac{9L(B + \sqrt{d}\epsilon + \Theta) \prod_{j=1}^L c_j}{\beta} \right)} \right\} \end{aligned}$$

Here $\alpha = \prod_{j=1}^L c_j (B + \sqrt{d}\epsilon)$. Take $\beta = \alpha / \sqrt{n}$, we have

$$\hat{\mathfrak{R}}_n(\ell \circ \tilde{\mathcal{F}}_{W_L}) \leq \hat{\mathfrak{R}}_n(\tilde{\mathcal{F}}_{W_L}) \leq \mathcal{O}$$

$$\left(\frac{\prod_{j=1}^L c_j (B + \sqrt{d}\epsilon)}{\sqrt{n}} \sqrt{d \sum_{j=1}^L r_j \ln \left(L \sqrt{n} \left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon} \right) \right)} \right).$$

Proof of Lemma 44.4.1

Besides the covering number of DNNs, Lemma 44.4.1 is obtained with the help of the covering number of Lipschitz function.

Lemma 44.6.5 *Given (z_1, \dots, z_n) from $\mathcal{Z} = \{z \in \mathbb{R}^r : \|z\| \leq A\}$, assume that $\tilde{\mathcal{H}}$ is a class of M -Lipschitz continuous functions mapping from \mathbb{R}^r to \mathbb{R} . Then, the covering number of $\tilde{\mathcal{H}}$ with respect to the metric ρ_∞ satisfies*

$$\mathcal{N}(\tau, \tilde{\mathcal{H}}, \rho_\infty) \leq \left(2 \left\lceil \frac{4MA}{\tau} \right\rceil + 1\right)^{\left(\frac{6MA}{\tau}\right)^r}.$$

Proof. We first scale $z \in \mathcal{Z}$ to be a unit ball, denoting as \mathcal{Z}/A for simplicity. For any $\tau_1 > 0$, there exists a τ_1 -covering of \mathcal{Z}/A consisting of $\mathcal{N}(\tau_1, \mathcal{Z}/A, \rho_\infty)$ balls: $\mathcal{B}_1, \dots, \mathcal{B}_{\mathcal{N}(\tau_1, \mathcal{Z}/A, \rho_\infty)}$. By [9], we have

$$\mathcal{N}(\tau_1, \mathcal{Z}/A, \rho_\infty) = \left(\frac{3}{\tau_1}\right)^r$$

Choose the center \mathbf{o}_t in each of the ball \mathcal{B}_t , for $t \in [\mathcal{N}(\tau_1, \mathcal{Z}/A, \rho_\infty)]$. The function $\tilde{h}(z)$ on the set \mathcal{Z} will be approximated by the construction:

$$\tilde{h}'(z) = \left\lceil \frac{2\tilde{h}(\mathbf{A}\mathbf{o}_t)}{\tau_2} \right\rceil \frac{\tau_2}{2} \text{ for } z \in \mathcal{B}_t$$

Take $\tau_1 = \tau_2/(2MA)$, we have

$$\begin{aligned} \sup_z |\tilde{h}(z) - \tilde{h}'(z)| &\leq \sup_z \left| \tilde{h}(z) - \frac{2\tilde{h}(\mathbf{A}\mathbf{o}_t)}{\tau_2} \frac{\tau_2}{2} \right| \\ &\quad + \sup_z \left| \frac{2\tilde{h}(\mathbf{A}\mathbf{o}_t)}{\tau_2} \frac{\tau_2}{2} - \left\lceil \frac{2\tilde{h}(\mathbf{A}\mathbf{o}_t)}{\tau_2} \right\rceil \frac{\tau_2}{2} \right| \\ &\leq \sup_z |\tilde{h}(z) - \tilde{h}(\mathbf{A}\mathbf{o}_t)| + \frac{\tau_2}{2} \\ &\leq M \sup_z \|z - \mathbf{A}\mathbf{o}_t\|_2 + \frac{\tau_2}{2} \\ &\leq MA \sup_z \|z/A - \mathbf{o}_t\|_2 + \frac{\tau_2}{2} \leq MA\tau_1 \\ &\quad + \frac{\tau_2}{2} \leq \tau_2 \end{aligned}$$

Let $s = 2\lceil 4AM/\tau_2 \rceil + 1$. The function $\tilde{h}'(z)$ assumes no more than $\mathcal{N}(\tau_1, \mathcal{Z}/A, \rho_\infty)$ values on each set s , and, therefore, the total number of all functions is no greater than the number $s^{\mathcal{N}(\tau_1, \mathcal{Z}/A, \rho_\infty)}$, that is,

$$\mathcal{N}(\tau_2, \tilde{\mathcal{H}}, \rho_\infty) \leq \left(2 \left\lceil \frac{4MA}{\tau_2} \right\rceil + 1\right)^{\left(\frac{6MA}{\tau_2}\right)^r}. \quad \square$$

Note that Lemma 44.6.1 is also applied for any fixed $l \in \{1, \dots, L\}$. Using Lemmas 44.6.1 and 44.6.5, we compute the Rademacher complexity for the decomposed DNN. First, we argue that for $\tau > 0$

$$\mathcal{N}(\tau, \ell \circ \mathcal{H}_{r_l} \circ \tilde{\mathcal{F}}_{\mathcal{W}_l}, \rho_n) \leq \mathcal{N}\left(\frac{\tau}{2}, \ell \circ \mathcal{H}_{r_l}, \rho_\infty\right) \mathcal{N}\left(\frac{\tau}{2 \prod_{j=l}^L c_j}, \tilde{\mathcal{F}}_{\mathcal{W}_l}, \rho_n\right).$$

Pick any function $\tilde{h}_{r_l} := \ell \circ h_{r_l} \in \ell \circ \mathcal{H}_{r_l}$ and $f_{\mathcal{W}_l} \in \tilde{\mathcal{F}}_{\mathcal{W}_l}$, and let \tilde{h}'_{r_l} and $f'_{\mathcal{W}_l}$ be the closest function in $\ell \circ \mathcal{H}_{r_l}$ and $\tilde{\mathcal{F}}_{\mathcal{W}_l}$, respectively. Since \tilde{h}'_{r_l} is $\prod_{j=l}^L c_j$ -Lipschitz, we have

$$\begin{aligned} \rho_n(\tilde{h}_{r_l} f_{\mathcal{W}_l}, \tilde{h}'_{r_l} f'_{\mathcal{W}_l}) &= \sqrt{\frac{1}{n} \sum_{i=1}^n |\tilde{h}_{r_l} f_{\mathcal{W}_l} - \tilde{h}'_{r_l} f'_{\mathcal{W}_l}|^2} \\ &\leq \sqrt{\frac{1}{n} \sum_{i=1}^n |\tilde{h}_{r_l} f_{\mathcal{W}_l} - \tilde{h}'_{r_l} f_{\mathcal{W}_l}|^2 + \frac{1}{n} \sum_{i=1}^n |\tilde{h}'_{r_l} f_{\mathcal{W}_l} - \tilde{h}'_{r_l} f'_{\mathcal{W}_l}|^2} \\ &\leq \sup_{\tilde{x}} |\tilde{h}_{r_l}(\tilde{x}) - \tilde{h}'_{r_l}(\tilde{x})| + \prod_{j=l}^L c_j \sqrt{\frac{1}{n} \sum_{i=1}^n \|f_{\mathcal{W}_l} - f'_{\mathcal{W}_l}\|_2^2} \\ &\leq \frac{\tau}{2} + \frac{\tau}{2} = \tau \end{aligned}$$

Therefore, we can choose \tilde{h}'_{r_l} and $f'_{\mathcal{W}_l}$ from the covers of $\ell \circ \mathcal{H}_{r_l}$ and $\tilde{\mathcal{F}}_{\mathcal{W}_l}$ to cover $\ell \circ \mathcal{H}_{r_l} \circ \tilde{\mathcal{F}}_{\mathcal{W}_l}$. By standard Dudley's entropy integral, we have

$$\begin{aligned}
\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \tilde{\mathcal{F}}_{\mathcal{W}_l}) &\lesssim \inf_{\beta > 0} \left\{ \beta + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{\ln \mathcal{N}(\tau, \ell \circ \mathcal{H}_{r_l} \circ \tilde{\mathcal{F}}_{\mathcal{W}_l}, \rho_n)} d\tau \right\} \\
&\leq \inf_{\beta > 0} \left\{ \beta + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{\ln \mathcal{N}\left(\frac{\tau}{2}, \ell \circ \mathcal{H}_{r_l}, \rho_{\infty}\right)} d\tau + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{\ln \mathcal{N}\left(\frac{\tau}{2 \prod_{j=1}^L c_j}, \tilde{\mathcal{F}}_{\mathcal{W}_l}, \rho_n\right)} d\tau \right\} \\
&\leq \inf_{\beta > 0} \left\{ \beta + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{\left(\frac{12(B + \sqrt{d}\epsilon) \prod_{j=1}^L c_j}{\tau}\right)^{r_l} \ln\left(2 \left\lceil \frac{8(B + \sqrt{d}\epsilon) \prod_{j=1}^L c_j}{\tau} \right\rceil + 1\right)} d\tau \right. \\
&\quad \left. + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{(2d+1) \sum_{j=1}^l r_j \ln\left(\frac{18 \prod_{j=1}^L c_j l (B + \sqrt{d}\epsilon + \Theta)}{\tau}\right)} d\tau \right\} \\
&:= \inf_{\beta > 0} \{P + Q\},
\end{aligned}$$

where

$$\alpha = \sup_{\substack{\mathbf{x} \in \mathcal{X} \\ \ell h_{r_l} f_{\mathcal{W}_l}(\mathbf{x}, 0) = (B + \sqrt{d}\epsilon) \prod_{j=1}^L c_j}} \rho_n(\ell h_{r_l} f_{\mathcal{W}_l}(\mathbf{x}, 0)) = (B + \sqrt{d}\epsilon) \prod_{j=1}^L c_j + \frac{16\alpha}{\sqrt{n}} \sqrt{\frac{2\pi}{2-r_l}} \mathbb{E}t^2 = \beta + \sqrt{\frac{2\pi}{m}} \frac{16\alpha}{(2-r_l)^{3/2}}$$

We consider Q ,

$$\begin{aligned}
Q &= \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{(2d+1) \sum_{j=1}^l r_j \ln\left(\frac{18 \prod_{j=1}^L c_j l (B + \sqrt{d}\epsilon + \Theta)}{\tau}\right)} d\tau \\
&\leq \frac{\alpha}{\sqrt{n}} \sqrt{(2d+1) \sum_{j=1}^l r_j \ln\left(\frac{18l(\alpha + \prod_{j=1}^L c_j \Theta)}{\beta}\right)}
\end{aligned}$$

Then we consider P ,

$$\begin{aligned}
P &= \beta + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{\left(\frac{12\alpha}{\tau}\right)^{r_l} \ln\left(2 \left\lceil \frac{8\alpha}{\tau} \right\rceil + 1\right)} d\tau \\
&\lesssim \beta + \frac{1}{\sqrt{n}} \int_{\beta}^{\alpha} \sqrt{\left(\frac{16\alpha}{\tau}\right)^{r_l} \ln\left(\frac{16\alpha}{\tau}\right)} d\tau \\
&\leq \beta + \frac{32\alpha}{\sqrt{n}} \int_{-\sqrt{\ln \frac{16\alpha}{\beta}}}^{-\sqrt{\ln 16}} e^{(r_l/2-1)t^2} t^2 dt
\end{aligned}$$

(a) When $r_l/2 - 1 < 0$, i.e., $r_l = 1$,

$$P \leq \beta + \frac{32\alpha}{\sqrt{n}} \int_{-\sqrt{\ln \frac{16\alpha}{\beta}}}^{-\sqrt{\ln 16}} e^{(r_l/2-1)t^2} t^2 dt \leq \beta$$

Therefore,

$$\begin{aligned}
P + Q &\leq \beta + \sqrt{\frac{2\pi}{m}} \frac{16\alpha}{(2-r_l)^{3/2}} + \frac{\alpha}{\sqrt{n}} \\
&\quad \sqrt{(2d+1) \sum_{j=1}^l r_j \ln\left(\frac{18l(\alpha + \prod_{j=1}^L c_j \Theta)}{\beta}\right)}
\end{aligned}$$

Take $\beta = \alpha/\sqrt{n}$, we have

$$\begin{aligned}
\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \tilde{\mathcal{F}}_{\mathcal{W}_l}) &\lesssim \frac{\alpha}{\sqrt{n}} \left(1 + 16\sqrt{2\pi}\right. \\
&\quad \left. + \sqrt{(2d+1) \sum_{j=1}^l r_j \ln\left(18l\sqrt{n}\left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon}\right)\right)}\right)
\end{aligned}$$

(b) When $r_l/2 - 1 = 0$, i.e., $r_l = 2$,

$$\begin{aligned}
P &\leq \beta + \frac{32\alpha}{\sqrt{n}} \int_{-\sqrt{\ln \frac{16\alpha}{\beta}}}^{-\sqrt{\ln 16}} t^2 dt = \beta + \frac{32\alpha}{3\sqrt{n}} \\
&\quad \left(\left(\ln \frac{16\alpha}{\beta}\right)^{3/2} - (\ln 16)^{3/2}\right)
\end{aligned}$$

Hence,

$$P + Q \leq \beta + \frac{32\alpha}{3\sqrt{n}} \left(\left(\ln \frac{16\alpha}{\beta} \right)^{3/2} - (\ln 16)^{3/2} \right) + \frac{\alpha}{\sqrt{n}} \sqrt{(2d+1) \sum_{j=1}^l r_j \ln \left(\frac{18l(\alpha + \prod_{j=1}^l c_j \Theta)}{\beta} \right)}$$

Choose $\beta = \alpha/\sqrt{n}$, we have

$$\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \widetilde{\mathcal{F}}_{\mathcal{W}_l}) \lesssim \frac{\alpha}{\sqrt{n}} \left(1 + 16 \left(\left(\ln(16\sqrt{n}) \right)^{3/2} - (\ln 16)^{3/2} \right) \right) + \frac{\alpha}{\sqrt{n}} \left(\sqrt{(2d+1) \sum_{j=1}^l r_j \ln \left(18l\sqrt{n} \left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon} \right) \right)} \right).$$

(c) When $r_l/2 - 1 > 0$, i.e., $r_l > 2$,

$$P \leq \beta + \frac{32\alpha}{\sqrt{n}} \int_{-\sqrt{\ln \frac{16\alpha}{\beta}}}^{-\sqrt{\ln 16}} e^{(r_l/2-1)t^2} t^2 dt \leq \beta + \frac{16\alpha}{r_l/2 - 1} \sqrt{\frac{\ln(16\alpha/\beta)}{m}} \left(\left(\frac{16\alpha}{\beta} \right)^{r_l/2-1} - 16^{r_l/2-1} \right)$$

Therefore,

$$P + Q \leq \beta + \frac{16\alpha}{r_l/2 - 1} \sqrt{\frac{\ln(16\alpha/\beta)}{m}} \left(\left(\frac{16\alpha}{\beta} \right)^{r_l/2-1} - 16^{r_l/2-1} \right) + \frac{\alpha}{\sqrt{n}} \sqrt{(2d+1) \sum_{j=1}^l r_j \ln \left(\frac{18l(\alpha + \prod_{j=1}^l c_j \Theta)}{\beta} \right)}$$

Let $\beta = \alpha/\sqrt[2]{m}$, we have

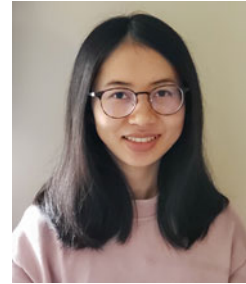
$$\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{r_l} \circ \widetilde{\mathcal{F}}_{\mathcal{W}_l}) \lesssim \frac{\alpha}{\sqrt[2]{n}} \left(1 + \frac{32}{r_l - 2} \sqrt{16^{r_l-2} \ln(16\sqrt[2]{n})} \right) + \frac{\alpha}{\sqrt[2]{n}} \left(\sqrt{(2d+1) \sum_{j=1}^l r_j \ln \left(18l\sqrt[2]{n} \left(1 + \frac{\Theta}{B + \sqrt{d}\epsilon} \right) \right)} \right).$$

This completes the proof.

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein gan. Preprint (2017). arXiv:1701.07875
2. Attias, I., Kontorovich, A., Mansour, Y.: Improved generalization bounds for robust learning (2018). arXiv:1810.02180
3. Bartlett, P., Shawe-Taylor, J.: Generalization performance of support vector machines and other pattern classifiers. In: Advances in Kernel Methods—Support Vector Learning, pp. 43–54 (1999)
4. Bartlett, P.L.: The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. IEEE Trans. Inf. Theory **44**(2), 525–536 (1998)
5. Bartlett, P.L.: Cs281b/stat241b: Statistical Learning Theory. University of California, Berkeley (2016)
6. Bartlett, P.L., Foster, D.J., Telgarsky, M.J.: Spectrally-normalized margin bounds for neural networks. In: Advances in Neural Information Processing Systems, pp. 6240–6249 (2017)
7. Biggio, B., Roli, F.: Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognition **84**, 317–331 (2017)
8. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2006)
9. Candes, E.J., Plan, Y.: Tight oracle bounds for low-rank matrix recovery from a minimal number of random measurements. Mathematics **57**(4), 2342–2359 (2010)
10. Cortes, C., Mohri, M., Rostamizadeh, A.: New generalization bounds for learning kernels. Preprint (2009). arXiv:0912.3309
11. Dalvi, N., Domingos, P., Mausam, Sanghai, S., Verma, D.: Adversarial classification. In: Tenth Acm Sigkdd International Conference on Knowledge Discovery & Data Mining (2004)
12. Farnia, F., Zhang, J.M., Tse, D.: Generalizable adversarial training via spectral normalization. Preprint (2018). arXiv:1811.07457
13. Golowich, N., Rakhlin, A., Shamir, O.: Size-independent sample complexity of neural networks. Preprint (2017). arXiv:1712.06541
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
15. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. Computer Science (2014)
16. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Networks **2**(5), 359–366 (1989)
17. Khim, J., Loh, P.-L.: Adversarial risk bounds for binary classification via function transformation. Preprint (2018). arXiv:1810.09519
18. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. Preprint (2013). arXiv:1312.6114
19. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. Preprint (2016). arXiv:1609.02907
20. Krogh, A., Hertz, J.A.: A simple weight decay can improve generalization. In: Advances in Neural Information Processing Systems, pp. 950–957 (1992)
21. Ledoux, M., Talagrand, M.: Probability in Banach Spaces: Isoperimetry and Processes. Springer Science & Business Media (2013)
22. Li, X., Lu, J., Wang, Z., Haupt, J., Zhao, T.: On tighter generalization bound for deep neural networks: Cnns, resnets, and beyond. Preprint (2018). arXiv:1806.05159
23. Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D.: Continuous control with deep reinforcement learning. Preprint (2015). arXiv:1509.02971

24. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks (2017). arXiv:1706.06083
25. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., et al.: Human-level control through deep reinforcement learning. *Nature* **518**(7540), 529 (2015)
26. Mohri, M., Rostamizadeh, A., Talwalkar, A.: *Foundations of Machine Learning* (2012).
27. Neyshabur, B., Bhojanapalli, S., Srebro, N.: A PAC-bayesian approach to spectrally-normalized margin bounds for neural networks. In: *International Conference on Learning Representations* (2018)
28. Neyshabur, B., Tomioka, R., Srebro, N.: Norm-based capacity control in neural networks. In: *Conference on Learning Theory*, pp. 1376–1401 (2015)
29. Raghunathan, A., Steinhardt, J., Liang, P.: Certified defenses against adversarial examples. Preprint (2018). arXiv:1801.09344
30. Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., Madry, A.: Adversarially robust generalization requires more data (2018). In: *32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*, Montreal, Canada
31. Shawe-Taylor, J., Bartlett, P.L., Williamson, R.C., Anthony, M.: Structural risk minimization over data-dependent hierarchies. *IEEE Trans. Inf. Theory* **44**(5), 1926–1940 (1998)
32. Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. Eprint (2014). Arxiv
33. Srebro, N., Ben-David, S.: Learning bounds for support vector machines with learned kernels. In: *International Conference on Computational Learning Theory*, pp. 169–183. Springer (2006)
34. Sun, S., Chen, W., Wang, L., Liu, X., Liu, T.-Y.: On the depth of deep neural networks: A theoretical view. In: *AAAI*, pp. 2066–2072 (2016)
35. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. *Computer Science* (2013)
36. Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., Madry, A.: There is no free lunch in adversarial robustness (but there are unexpected benefits) (2018)
37. Vapnik, V.: *The Nature of Statistical Learning Theory*. Springer Science & Business Media (2013)
38. Vapnik, V., Chappelle, O.: Bounds on error expectation for support vector machines. *Neural Computation* **12**(9), 2013–2036 (2000)
39. Xu, Y., Wang, X.: Understanding weight normalized deep neural networks with rectified linear units. In: *Advances in Neural Information Processing Systems*, pp. 130–139 (2018)
40. Yin, D., Ramchandran, K., Bartlett, P.: Rademacher complexity for adversarially robust generalization. Preprint (2018). arXiv:1810.11914
41. Yoshida, Y., Miyato, T.: Spectral norm regularization for improving the generalizability of deep learning. Preprint (2017). arXiv:1705.10941



Qingyi Gao is a Ph.D. candidate at Purdue University. Ms. Gao received her B.S. and M.S. from Nankai University in China. Ms. Gao works on deep learning theory, generative models, and adversarial learning.



Xiao Wang received his Ph.D. from University of Michigan in Statistics in 2005. He worked at University of Maryland Baltimore County and Purdue University. He is now full professor at Purdue University. Dr. Wang works on machine learning, deep learning, nonparametric statistics, and functional data analysis. He is well known for working both on theory and applications.



Bayesian Survival Analysis in the Presence of Monotone Likelihoods

45

Jing Wu , Mário de Castro , and Ming-Hui Chen

Contents

45.1	Introduction	921
45.2	Models, Examples, and Monotone Likelihood	922
45.2.1	Preliminary.....	922
45.2.2	Piecewise Constant Hazards Regression Model.....	922
45.2.3	Cox Proportional Hazards Regression Model.....	922
45.2.4	Illustrative Examples.....	923
45.2.5	Monotone Likelihood.....	924
45.3	Bayesian Inference	924
45.3.1	Prior and Posterior Distributions.....	924
45.3.2	Conditions for Posterior Propriety.....	925
45.3.3	Remedies for Posterior Propriety.....	925
45.4	Analysis of the Test Data from a Tire Reliability Study	925
45.5	Discussion	928
45.5.1	Alternative Prior Distributions.....	928
45.5.2	Bayesian Model Comparison.....	929
45.5.3	Software.....	929
References	929

can be divided into two groups. If the event of interest does not occur (zero event) for all subjects in one of the groups, the resulting likelihood function is monotonic and consequently the covariate effects are difficult to estimate. In this chapter, we carry out an in-depth examination of the conditions of the monotone likelihood problem under a parametric regression model and the partial likelihood under the Cox proportional hazards regression model. We review and discuss Bayesian approaches to handle the monotone likelihood and partial likelihood problems. We analyze the test data from a tire reliability study in details.

Keywords

Cox proportional hazards regression model · DIC · Monotone partial likelihood · Penalized maximum likelihood · Piecewise constant hazards regression model · Reliability analysis · WAIC · Zero events

Abstract

The monotone likelihood problem is often encountered in the analysis of time-to-event data under a parametric regression model or a Cox proportional hazards regression model when the sample size is small or the events are rare. For example, with a binary covariate, the subjects

45.1 Introduction

In [27], an overview of many aspects of the semiparametric proportional hazards regression model of [6] is provided. The overview covers parameter estimation, hypothesis testing, estimation of the survival and cumulative hazard functions, goodness-of-fit and model checking, and extensions to accommodate stratified data and time-dependent covariates. The authors also deal with models including random effects, nonproportional hazards, and multivariate survival data. An analysis of a data set on fatigue limit of two steel specimens is given as illustration. All the methods discussed in [27] are based on a frequentist (classic) point of view.

In many studies involving time-to-event data, it is often the case that experimental units within at least one group of the study will experience very few events or no event at all. This could be due to the length of the study or due to

J. Wu (✉)

Department of Computer Science and Statistics, University of Rhode Island, Kingston, RI, USA
e-mail: jing_wu@uri.edu

M. de Castro

Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, Brazil
e-mail: mcastro@icmc.usp.br

Ming-Hui Chen

Department of Statistics, University of Connecticut, Storrs, CT, USA
e-mail: ming-hui.chen@uconn.edu

the nature of the study itself. For example, among the subset of the Surveillance, Epidemiology, and End Results (SEER) program prostate cancer data in Example 3, no patients who receive surgery treatment die. If the interest is to analyze the surgery treatment effect on the time-to-event data in Example 3, then the nonoccurrence of events in the surgery treatment group will lead to a model identifiability issue.

We investigate the monotone likelihood problem when fitting the piecewise constant hazards regression and Cox proportional hazards regression models. Although we mainly focus on these two models, the monotone likelihood problem may also exist under different models such as accelerated failure time models [17, Ch. 7], parametric hazards models, and so on.

This chapter is organized as follows. In Sect. 45.2, we describe the monotone likelihood problem and give some examples. Section 45.3 is dedicated to Bayesian inference, including prior and posterior distributions as well as conditions for posterior propriety. In Sect. 45.4, we report results from the analysis of data on a tire reliability study. Alternative prior distributions, model comparison criteria, and software for fitting the Cox proportional hazards regression model are discussed in Sect. 45.5.

45.2 Models, Examples, and Monotone Likelihood

45.2.1 Preliminary

Let y_i denote the minimum of the censoring time C_i and the survival time T_i , and let $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})'$ be the $p \times 1$ vector of covariates associated with y_i for the i th subject. Denote by $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)'$ the $p \times 1$ vector of regression coefficients. Also, $v_i = 1\{T_i = y_i\}$ is the failure indicator for $i = 1, \dots, n$, where n is the total number of subjects and $\mathcal{R}(t) = \{i : y_i \geq t\}$ is the set of subjects at risk at time t . We assume throughout this chapter that $\boldsymbol{\beta}$ does not include an intercept and that given \mathbf{x}_i , T_i , and C_i are independent. Let $\mathcal{D}_{\text{obs}} = \{(y_i, v_i, \mathbf{x}_i) : i = 1, \dots, n\}$ denote the observed right censored data. Also write $\mathbf{y} = (y_1, \dots, y_n)'$ and $\mathbf{v} = (v_1, \dots, v_n)'$.

45.2.2 Piecewise Constant Hazards Regression Model

As discussed in [14, Ch. 3], one of the most popular models for semiparametric survival or reliability analysis is the piecewise constant hazards regression model. This model is constructed as follows. We first let $0 = s_0 < s_1 < s_2 < \dots < s_J = \infty$ denote a finite partition of the time axis. Then, we have J intervals $(0, s_1], (s_1, s_2], \dots, (s_{J-1}, \infty)$. In the j th interval, we assume a constant baseline hazard $h_0(y) = \lambda_j$,

for $y \in (s_{j-1}, s_j]$. Letting $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_J)'$, the likelihood function of $(\boldsymbol{\beta}, \boldsymbol{\lambda})$ is given by

$$L(\boldsymbol{\beta}, \boldsymbol{\lambda} | \mathcal{D}_{\text{obs}}) = \prod_{i=1}^n \prod_{j=1}^J \{ \lambda_j \exp(\mathbf{x}'_i \boldsymbol{\beta}) \}^{\delta_{ij} v_i} \times \exp \left[- \delta_{ij} \left\{ \lambda_j (y_i - s_{j-1}) + \sum_{g=1}^{j-1} \lambda_g (s_g - s_{g-1}) \right\} \exp(\mathbf{x}'_i \boldsymbol{\beta}) \right], \quad (45.1)$$

where $\delta_{ij} = 1$ if the i th subject failed or was censored in the j th interval and 0 otherwise. Define $\Delta_j(t)$ as follows:

$$\Delta_j(t) = \begin{cases} 0, & \text{if } t < s_{j-1}, \\ t - s_{j-1}, & \text{if } s_{j-1} \leq t < s_j, \\ s_j - s_{j-1}, & \text{if } t \geq s_j. \end{cases} \quad (45.2)$$

$L(\boldsymbol{\beta}, \boldsymbol{\lambda} | \mathcal{D}_{\text{obs}})$ in (45.1) can also be written as

$$L(\boldsymbol{\beta}, \boldsymbol{\lambda} | \mathcal{D}_{\text{obs}}) = \prod_{i=1}^n \{ \exp(\mathbf{x}'_i \boldsymbol{\beta}) \}^{v_i} \prod_{j=1}^J \lambda_j^{\sum_{i=1}^n \delta_{ij} v_i} \exp \left\{ - \sum_{j=1}^J \sum_{i=1}^n \Delta_j(y_i) \exp(\mathbf{x}'_i \boldsymbol{\beta}) \lambda_j \right\}. \quad (45.3)$$

Assume $\sum_{i=1}^n \delta_{ij} v_i > 0$, for $j = 1, \dots, J$. After plugging in the profile maximum likelihood estimator of $\boldsymbol{\lambda}$ in (45.3), the profile likelihood function is given by

$$L_{pf}(\boldsymbol{\beta} | \mathcal{D}_{\text{obs}}) = \prod_{i=1}^n \left\{ \frac{\exp(\mathbf{x}'_i \boldsymbol{\beta})}{\sum_{j=1}^J \Delta_j(y_i) \exp(\mathbf{x}'_i \boldsymbol{\beta})} \right\}^{v_i} \times \exp \left(- \sum_{i=1}^n v_i \right) \prod_{j=1}^J \left(\sum_{i=1}^n \delta_{ij} v_i \right)^{\sum_{i=1}^n \delta_{ij} v_i}, \quad (45.4)$$

where j_i is the interval in which the i th subject failed or was censored.

45.2.3 Cox Proportional Hazards Regression Model

The likelihood function under the Cox proportional hazards regression model [7] is given by

$$L(\boldsymbol{\beta}, h_0 | \mathcal{D}_{\text{obs}}) = \prod_{i=1}^n \{ h_0(y_i) \exp(\mathbf{x}'_i \boldsymbol{\beta}) \}^{v_i} \exp \{ - H_0(y_i) \exp(\mathbf{x}'_i \boldsymbol{\beta}) \}, \quad (45.5)$$

where $H_0(t) = \int_0^t h_0(u)du$ is the cumulative baseline hazard function. Then, the partial likelihood function is given by

$$L_p(\boldsymbol{\beta}|\mathcal{D}_{\text{obs}}) = \prod_{i=1}^n \left\{ \frac{\exp(\mathbf{x}'_i \boldsymbol{\beta})}{\sum_{j \in \mathcal{R}(y_i)} \exp(\mathbf{x}'_j \boldsymbol{\beta})} \right\}^{v_i}. \quad (45.6)$$

45.2.4 Illustrative Examples

In this section, we provide a few examples in which the monotone likelihood or the monotone partial likelihood problem arise.

Example 1 Consider a survival data set with $n = 3$ subjects, $p = 2$ covariates, $\mathbf{x}_1 = (0, 1)'$, $\mathbf{x}_2 = (1, 0)'$, $\mathbf{x}_3 = (0, 1)'$, $\mathbf{v} = (1, 0, 0)'$, and $(y_1, y_2, y_3)' = (1, 1, 1)'$. In this case, $\mathcal{R}(y_1) = \{1, 2, 3\}$. Assuming $J = 1$ and letting $\lambda = \lambda_1$, the likelihood function in (45.1) takes the form

$$L(\boldsymbol{\beta}, \lambda|\mathcal{D}_{\text{obs}}) = \lambda \exp(\beta_2) \exp \left[-\lambda \{ \exp(\beta_1) + 2 \exp(\beta_2) \} \right]. \quad (45.7)$$

Thus, $L(\boldsymbol{\beta}, \lambda|\mathcal{D}_{\text{obs}})$ is monotonic in β_1 . The partial likelihood function in (45.6) is given by

$$L_p(\boldsymbol{\beta}|\mathcal{D}_{\text{obs}}) = \exp(\beta_2) / \{ \exp(\beta_1) + 2 \exp(\beta_2) \}. \quad (45.8)$$

It is easy to see that $L_p(\boldsymbol{\beta}|\mathcal{D}_{\text{obs}})$ is a monotonic function of β_1 or β_2 , as displayed in Fig. 45.1. The maximum of $L_p(\boldsymbol{\beta}|\mathcal{D}_{\text{obs}})$ is attained when $\beta_1 - \beta_2 \rightarrow -\infty$. It is interesting to mention that after profiling λ out in (45.7), the profile likelihood function is given by

$$L_{pf}(\boldsymbol{\beta}|\mathcal{D}_{\text{obs}}) = \exp(\beta_2 - 1) / \{ \exp(\beta_1) + 2 \exp(\beta_2) \},$$

which is proportional to $L_p(\boldsymbol{\beta}|\mathcal{D}_{\text{obs}})$ given in (45.8).

Example 2 Chen et al. [5] considered data, which consist of $n = 550$ men who were treated with radiation therapy between 1989 and 2002 following with 6 months of with short-course androgen suppression therapy for localized prostate cancer with at least one adverse risk factor represented by prostate-specific antigen (PSA) > 10 ng/ml, biopsy Gleason score 7 to 10, or 2002 American Joint Commission on Cancer (AJCC) clinical tumor category T2b or T2c. The outcome variable (y_i) in years was time to prostate cancer death, which is continuous and subject to right censoring, and $v_i = 1$ if the i th subject died due to prostate cancer and 0 otherwise. Define $A = 1\{\text{PSA} > 10\}$, $B = 1\{\text{Gleason} \geq 7\}$, and $C = 1\{\text{T2b or T2c}\}$. The Cox proportional hazards model with covariates $A \times B$, $A \times C$, $B \times C$, $A \times B \times C$, and age (in years)

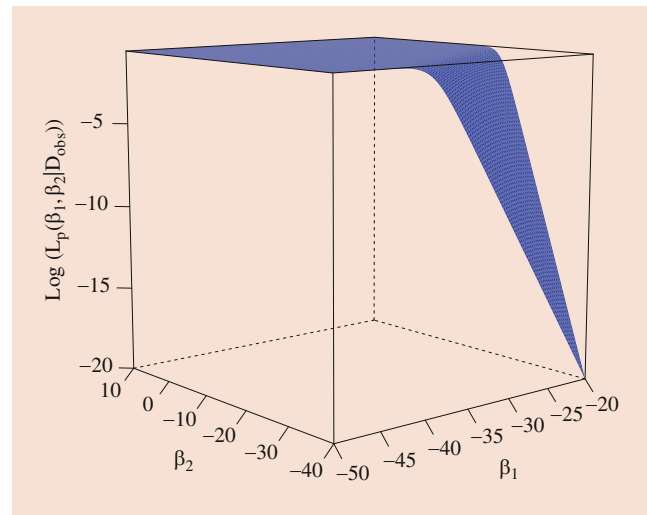


Fig. 45.1 Log partial likelihood function for $(\beta_1, \beta_2)'$ in Example 1

Table 45.1 Output from SAS PHREG procedure—Example 2

Variable	DF	Parameter Estimate	Standard Error	Chi-square	Pr > ChiSq
A×B	1	0.39759	1.23355	0.1039	0.7472
A×C	1	-14.30314	2107	0.0000	0.9946
B×C	1	0.59060	1.22714	0.2316	0.6303
A×B×C	1	2.22155	0.80450	7.6253	0.0058
Age	1	0.02262	0.04821	0.2201	0.6390

was fitted to this data set and the results are displayed in Table 45.1.

From Table 45.1, we see that the absolute value of the estimate of the regression coefficient corresponding to $A \times C$ was large and the corresponding standard error was huge, which certainly implies a monotone partial likelihood problem.

Example 3 Wu et al. [29] considered 1840 men subjects from the SEER prostate cancer data between 1973 to 2013, who had all of the three intermediate risk factors: clinical tumor stage is T2b or T2c, Gleason score equals 7, and PSA level between 10 and 20 ng/ml. Among those 1840 subjects, the total number of events due to prostate cancer was 8. The covariates considered in the analysis were PSA level, surgery treatment indicator (RP), radiation treatment only indicator (RT), African-American indicator (Black), year of diagnosis (Year_diag), and age (in years). The Cox proportional hazards model was fitted to study the death due to prostate cancer. The resulting maximum partial likelihood estimates (MPLEs) are shown in Table 45.2. Note that in Table 45.2, N_c represents the number of censored, and N_{pc} represents the number of prostate cancer deaths. We see from Table 45.2 that for RP, the MPLE (Est) and the standard error (SE) were -17.745 and 1680, respectively. These results indicate that

Table 45.2 Maximum partial likelihood estimates for the prostate cancer data—Example 3

Variable	N_c	N_{pc}	Est	SE	p -value
PSA	1840		0.253	0.468	0.5881
RP	842	0	-17.745	1680	0.9916
RT	576	3	-1.150	0.742	0.1210
Black	279	1	-0.539	1.125	0.6318
Year_diag	1840		-0.377	0.743	0.6118
Age	1840		-0.372	0.416	0.3712

RP is not identifiable for the death caused by prostate cancer, which is due to the absence of events (prostate cancer death) in the “surgery treatment” group of patients.

45.2.5 Monotone Likelihood

Piecewise Constant Hazards Regression Model

Note that the profile likelihood function in (45.4) is log-concave with respect to β . Here, we give a general definition of the monotone likelihood problem for the piecewise constant hazards regression model: the profile likelihood function in (45.4) converges to a finite value when at least one parameter estimate goes to $-\infty$ or $+\infty$, and thus the maximum likelihood estimate (MLE) of the corresponding parameter does not exist.

Cox Proportional Hazards Regression Model

Similarly, the monotone likelihood problem for the Cox proportional hazards regression model is defined as follows: the

partial likelihood function in (45.6) converges to a finite value when at least one parameter estimate goes to $-\infty$ or $+\infty$, and thus the MPLE does not exist. For the Cox proportional hazards model, Bryson and Johnson [2] pointed out that for any sample size, there is a non-null probability that the MPLE of β will be infinite. Quoting Heinze and Ploner [12], “The probability of occurrence of monotone likelihood is too high to be negligible.”

45.3 Bayesian Inference

45.3.1 Prior and Posterior Distributions

Piecewise Constant Hazards Regression Model

Let $\pi(\beta)$ and $\pi(\lambda)$ denote the prior distributions for β and λ , respectively. To construct the joint prior for (β, λ) , we assume that β and λ_j are independent, for $j = 1, \dots, J$. We further assume a Jeffreys-type prior for $\lambda_j, j = 1, \dots, J$. Then, the joint prior is specified as

$$\pi(\beta, \lambda) = \pi(\beta) \prod_{j=1}^J \pi(\lambda_j) = \pi(\beta) \prod_{j=1}^J 1/\lambda_j. \quad (45.9)$$

By combining the likelihood function in (45.1) with the prior distribution in (45.9), the posterior distribution for (β, λ) is then given by

$$\begin{aligned} \pi(\beta, \lambda | \mathcal{D}_{\text{obs}}) &\propto \pi(\beta, \lambda) L(\beta, \lambda | \mathcal{D}_{\text{obs}}) = \pi(\beta) \prod_{j=1}^J 1/\lambda_j \prod_{i=1}^n \prod_{j=1}^J \{\lambda_j \exp(\mathbf{x}'_i \beta)\}^{\delta_{ij} v_i} \\ &\quad \times \exp \left[-\delta_{ij} \left\{ \lambda_j (y_i - s_{j-1}) + \sum_{g=1}^{j-1} \lambda_g (s_g - s_{g-1}) \right\} \exp(\mathbf{x}'_i \beta) \right] \\ &= \pi(\beta) \prod_{i=1}^n \left\{ \exp(\mathbf{x}'_i \beta)^{v_i} \right\} \prod_{j=1}^J \lambda_j^{\sum_{i=1}^n \delta_{ij} v_i - 1} \exp \left\{ -\sum_{j=1}^J \sum_{i=1}^n \Delta_j(y_i) \exp(\mathbf{x}'_i \beta) \lambda_j \right\}. \end{aligned} \quad (45.10)$$

If we further assume $\pi(\beta) \propto 1$ in (45.9), the posterior distribution in (45.10), after integrating out λ , is proportional to the profile likelihood function in (45.4).

Cox Proportional Hazards Regression Model

Kalbfleisch [16] and Sinha et al. [25] showed that the partial likelihood function in (45.6) can be obtained as a limiting case of the marginal posterior distribution of β with continuous time survival data under a gamma process prior for the cumulative baseline hazard function $H_0(\cdot)$ using the

likelihood function in (45.5). If we treat the partial likelihood $L_p(\beta | \mathcal{D}_{\text{obs}})$ in (45.6) as the likelihood function, the posterior distribution for β is then given by

$$\begin{aligned} \pi(\beta | \mathcal{D}_{\text{obs}}) &\propto \pi(\beta) L_p(\beta | \mathcal{D}_{\text{obs}}) = \pi(\beta) \\ &\quad \times \prod_{i=1}^n \left\{ \frac{\exp(\mathbf{x}'_i \beta)}{\sum_{j \in \mathcal{R}(y_i)} \exp(\mathbf{x}'_j \beta)} \right\}^{v_i}. \end{aligned} \quad (45.11)$$

45.3.2 Conditions for Posterior Propriety

Piecewise Constant Hazards Regression Model

Note that we can only carry out Bayesian inference when the posterior distribution is proper, i.e., the integral of the right-hand side in (45.10) over the parameter space is finite. Ge and Chen [9] proved that with the joint prior of $(\boldsymbol{\beta}, \boldsymbol{\lambda})$ specified in (45.9) and a uniform prior for $\boldsymbol{\beta}$, the posterior distribution (45.10) is proper, if conditions (i) $y_i > 0$, for $i = 1, \dots, n$, and (ii) $[v_i(\delta_{i1}, \dots, \delta_{ij}, \mathbf{x}'_i)]'$, $i = 1, \dots, n$ is of full column rank are satisfied. Condition (i) essentially requires that all event times are strictly positive. Condition (ii) requires that the corresponding matrix is of full rank and also implies that at least one event occurs in each interval, i.e., $\sum_{i=1}^n \delta_{ij} v_i > 0$, for $j = 1, \dots, J$.

Cox Proportional Hazards Regression Model

Define \mathbf{X}^* to be the matrix given by

$$\mathbf{X}^* = [v_i(\mathbf{x}_j - \mathbf{x}_i) : j \in \mathcal{R}(y_i), i = 1, \dots, n]'. \quad (45.12)$$

The necessary and sufficient conditions established in [4] for propriety of the posterior when an improper uniform prior is assumed for $\boldsymbol{\beta}$ are given by

- C1. The matrix \mathbf{X}^* is of full column rank and
- C2. There exists a positive vector $\boldsymbol{\nu}$ such that $\mathbf{X}^{*\prime} \boldsymbol{\nu} = \mathbf{0}$. (45.13)

A positive vector $\boldsymbol{\nu}$ means that each component of $\boldsymbol{\nu}$ is positive. Condition C2 can be checked by solving a linear programming problem [23].

Here we consider Example 1 again, which does not satisfy C1 and C2. Thus, the MPLE of $\boldsymbol{\beta}$ does not exist, and consequently, the posterior distribution is improper.

Example 1 (continuation) In this case, $\mathcal{R}(y_1) = \{1, 2, 3\}$ and \mathbf{X}^* in (45.12) has rows $(0, 0)$, $(1, -1)$, and $(0, 0)$, so that $\text{rank}(\mathbf{X}^*) = 1$ and condition C1 in (45.13) breaks down. The partial likelihood function in (45.6) is given by $L_p(\boldsymbol{\beta} | \mathcal{D}_{\text{obs}}) = \exp(\beta_2) / \{\exp(\beta_1) + 2 \exp(\beta_2)\}$. The maximum of $L_p(\boldsymbol{\beta} | \mathcal{D}_{\text{obs}})$ is attained when $\beta_1 - \beta_2 \rightarrow -\infty$.

45.3.3 Remedies for Posterior Propriety

If conditions in Sect. 45.3.2 for the piecewise constant hazards regression model and conditions in Sect. 45.3.2 for the Cox proportional hazards model are not satisfied, the posterior distribution under improper priors for $\boldsymbol{\beta}$ will be improper. It thus becomes infeasible to carry out Bayesian inference. Note that the posterior distribution is always proper under a

proper prior. Therefore, in order to guarantee the posterior propriety, a proper prior need to be assumed for $\boldsymbol{\beta}$. Below, we introduce one of the most popular proper priors for $\boldsymbol{\beta}$, namely, the Zellner's g -prior [30],

$$\boldsymbol{\beta} \sim N(\mathbf{0}, (g\tilde{\mathbf{X}}'\tilde{\mathbf{X}})^{-1}),$$

where $\tilde{\mathbf{X}}$ is the design matrix and g is either a constant or a hyperparameter, which is a parameter of a prior distribution. Note that a larger value of g corresponds to a more informative prior. More discussions about other proper priors for $\boldsymbol{\beta}$ are given in Sect. 45.5.1.

In Sect. 45.3.2, for the piecewise constant hazard regression model, we assume an improper Jeffreys-type prior for λ_j , for $j = 1, \dots, J$. However, if there are not many events in certain intervals, we need to assume a proper prior for the corresponding hazard parameter to guarantee the posterior propriety.

45.4 Analysis of the Test Data from a Tire Reliability Study

We consider the same test data set in [18] to study the time to failure of automobile tire due to tread and belt separation (TBS). Same as in [18], time to TBS failure was used as the response variable. Covariates considered include wedge gauge indicator (Wedge = 1 if wedge gauge is greater than the median and 0 otherwise), peel force indicator (Peel = 1 if peel force is greater than the median and 0 otherwise), the interaction between wedge gauge and peel force indicators (Wedge \times Peel), tire age (Tireage), interbelt gauge (Interbelt), end of belt #2 to buttress (Eb2b), and percent of carbon black (Crbnblk). Except for Wedge, Peel, and Wedge \times Peel, which are binary, all other covariates are continuous. The total number of tires used in the analysis was 34 and the number of TBS failure is 11. The continuous covariates were standardized for numerical stability.

The Cox proportional hazards model was fitted to study the effects of factors on time to TBS failure. The resulting MPLEs are shown in Table 45.3. We see from Table 45.3 that for Wedge \times Peel, the MPLE (Est) and the standard error (SE) were -19.44641 and 2997 , respectively (see also Fig. 45.2). These results indicate that the interaction term is not identifiable. One solution to the monotone likelihood problem from the frequentist point of view is to add the Firth's penalty to the likelihood function [13]. The detailed formulations of the Firth's penalty approach are given in Sect. 45.5.3. Table 45.4 presents the SAS outputs using the Firth's penalty. Wedge \times Peel is now identifiable with $\text{SE} = 2.69102$. Tireage with p -value = 0.0294 and Eb2b with p -value = 0.0356 were both significant.

Table 45.3 Output for the Cox proportional hazards regression model from SAS PHREG procedure—tire reliability study

Variable	DF	Parameter Estimate	Standard Error	Chi-square	Pr > ChiSq
Wedge	1	-0.79045	1.03396	0.5845	0.4446
Peel	1	0.33247	0.90740	0.1342	0.7141
Wedge × Peel	1	-19.44641	2997	0.0000	0.9948
Tireage	1	2.27269	0.97057	5.4831	0.0192
Interbelt	1	-0.17082	0.40221	0.1804	0.6711
Eb2b	1	-1.20412	0.51820	5.3994	0.0201
Crnblk	1	0.33732	0.36852	0.8378	0.3600

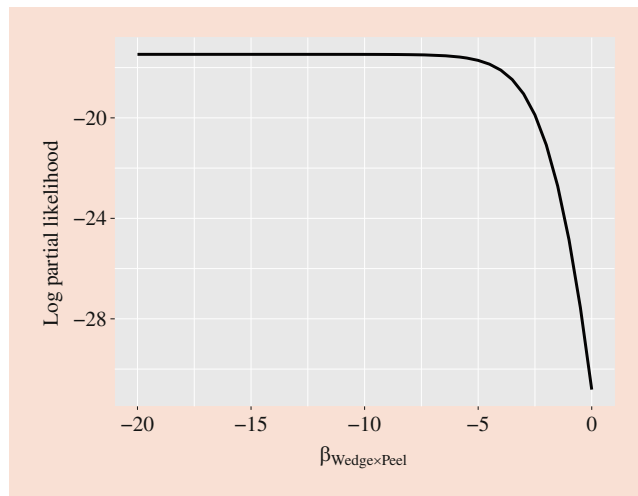


Fig. 45.2 Log partial likelihood function for $\beta_{\text{Wedge} \times \text{Peel}}$ in the tire reliability study

Table 45.4 Output for the Cox proportional hazards regression model from SAS PHREG procedure Using the Firth’s Penalty—tire reliability study

Variable	DF	Parameter Estimate	Standard Error	Chi-square	Pr > ChiSq
Wedge	1	-0.63997	0.99544	0.4133	0.5203
Peel	1	0.35633	0.90299	0.1557	0.6931
Wedge × Peel	1	-4.19009	2.69102	2.4244	0.1195
Tireage	1	2.00047	0.91872	4.7413	0.0294
Interbelt	1	-0.13587	0.38864	0.1222	0.7266
Eb2b	1	-0.99044	0.47128	4.4166	0.0356
Crnblk	1	0.30669	0.36249	0.7158	0.3975

To solve this problem from the Bayesian perspective, we first fit the tire data using the Cox proportional hazards regression model under various Zellner’s g -priors for β . In all the Bayesian computations, we used 10,000 Markov chain Monte Carlo (MCMC) samples, after a burn-in of 2000 iterations for each model to compute all posterior estimates, including posterior means, posterior standard deviations

(Standard Deviation), and 95% highest posterior density (HPD) intervals. All the analyses were done using SAS procedure PROC PHREG [24, Ch. 73]. The deviance information criterion (DIC) [26] was used to guide the choice of g in the Zellner’s g -prior. Based on the DIC criterion, the model with the smallest DIC value is the most optimal among all the models under consideration. As shown in Fig. 45.3, the value of DIC has an increasing trend as g increases from 0.001 to 0.1 with increment equals 0.001 and the value of the effective number of parameters (p_D) decreases as g increases. DIC attained the local minimum with DIC = 49.791 at $g = 0.001$ among all the models under consideration. Therefore, based on the DIC criteria, the model under the Zellner’s g -priors with $g = 0.001$ is the most optimal among all the models under consideration.

Table 45.5 shows the number of MCMC samples (N), the posterior mean, the posterior standard deviation, and 95% HPD intervals under the Zellner’s g -priors with $g = 0.001, 0.01, \text{ and } 0.1$. We took a posterior estimate to be “statistically significant at a significance level of 0.05” if the corresponding 95% HPD interval does not contain 0. The three separate rows under each coefficient are outputs for $g = 0.001, 0.01, \text{ and } 0.1$, respectively. Based on the posterior summaries in Table 45.5 for $g = 0.001$, Tireage with 95% HPD interval = (0.1801, 4.2824) and Eb2b with 95% HPD interval = (-2.6131, -0.3745) were significant. These results were similar to the results of the Firth’s penalty approach in Table 45.4. As shown in Table 45.5, posterior summaries were different under different Zellner’s g -priors. Eb2b was still significant for $g = 0.01$ and 0.1, while Tireage was not. We further note that the standard deviations of each parameter decreased as g increased.

Next, we fit the tire data using the piecewise constant hazards regression model. The Zellner’s g -priors with $g = 0.001, 0.01, \text{ and } 0.1$ were used for β , and the non-informative Jeffreys-type prior in Sect. 45.3.1 was assumed for λ . Due to the small number of events (TBS failure), we assume $J = 2$ intervals for the partition of the time axis to ensure that λ is identifiable. Under this joint prior, SAS PHREG procedure failed to provide output and reported “floating point overflow” error. We then ran the model using WinBUGS [19], which is another Bayesian software. The values of DIC (p_D) under the Zellner’s g -priors with $g = 0.001, 0.01, \text{ and } 0.1$ were 106.4 ($p_D = 8.5$), 106.7 ($p_D = 8.4$), 106.6 ($p_D = 6.6$), respectively. Based on the DIC criterion, the Zellner’s g -priors with $g = 0.001$ is still the most optimal among all the three priors under consideration. Note that the values of DIC from the piecewise constant hazards regression and Cox proportional hazards models are not comparable since the two models do not have the same number of terms.

Posterior summaries under the Zellner’s g -prior with $g = 0.001, 0.01, \text{ and } 0.1$ are presented in Table 45.6. Note that



Fig. 45.3 Plots of values of (a) DIC and (b) p_D

Table 45.5 Bayesian output for the Cox proportional hazards regression model from SAS PHREG procedure under the Zellner’s g -priors with $g = 0.001, 0.01, \text{ and } 0.1$ —tire reliability study

$g = 0.001/0.01/0.1$				
Parameter	N	Mean	Standard Deviation	95% HPD Interval
Wedge	10000	-0.8746	1.0944	(-3.0335, 1.2130)
	10000	-0.7447	1.0111	(-2.6837, 1.2758)
	10000	-0.4476	0.7816	(-1.9775, 1.0920)
Peel	10000	0.3266	1.0225	(-1.6347, 2.3533)
	10000	0.4201	0.9863	(-1.6116, 2.2727)
	10000	0.4057	0.8475	(-1.3320, 1.9741)
Wedge × Peel	10000	-8.6207	4.9396	(-18.8437, 0.1841)
	10000	-2.6180	2.2867	(-7.0552, 1.9422)
	10000	-0.4673	1.3798	(-3.2587, 2.1612)
Tireage	10000	2.2075	1.0463	(0.1801, 4.2824)
	10000	1.2653	0.7918	(-0.3357, 2.7553)
	10000	0.3534	0.4366	(-0.5179, 1.2081)
Interbelt	10000	-0.2972	0.4408	(-1.1655, 0.5691)
	10000	-0.4159	0.4046	(-1.2167, 0.3704)
	10000	-0.3412	0.3022	(-0.9347, 0.2313)
Eb2b	10000	-1.4398	0.5770	(-2.6131, -0.3745)
	10000	-1.3336	0.5383	(-2.4116, -0.2970)
	10000	-0.8725	0.3665	(-1.5901, -0.1560)
Crnbblk	10000	0.3104	0.3987	(-0.4904, 1.0911)
	10000	0.1628	0.3744	(-0.5647, 0.9156)
	10000	0.0226	0.2989	(-0.5533, 0.6124)

WinBUGS reports 95% credible intervals based on quantiles rather than 95% HPD intervals. Similar to the HPD intervals, a posterior estimate is considered “statistically significant at a significance level of 0.05” if the corresponding 95% credible interval does not contain 0. As shown in Table 45.6 for $g = 0.001$, Eb2b was significant with 95% credible

interval = (-2.916, -0.471). Similar to the Cox proportional hazards model, posterior summaries of the piecewise constant hazard model were not robust under different Zellner’s g -priors. Eb2b was still significant for $g = 0.01$ and 0.1 , and the standard deviations of each parameter decreased as g increased.

Table 45.6 Bayesian output for the piecewise constant hazards regression model from WinBUGS under the Zellner's g -priors with $g = 0.001, 0.01, \text{ and } 0.1$ —tire reliability study

$g=0.001/0.01/0.1$							
Parameter	Mean	Standard Deviation	2.5%	25%	50%	75%	97.5%
Wedge	-0.371	0.952	-2.326	-0.949	-0.342	0.295	1.349
	-0.390	0.912	-2.318	-0.945	-0.377	0.267	1.305
	-0.278	0.740	-1.854	-0.754	-0.240	0.214	1.086
Peel	0.027	0.921	-1.929	-0.550	0.105	0.668	1.710
	0.120	0.946	-1.881	-0.467	0.198	0.750	1.783
	0.137	0.816	-1.658	-0.385	0.174	0.714	1.593
Wedge×Peel	-7.389	5.534	-20.215	-10.612	-6.319	-3.312	0.877
	-2.123	2.135	-6.609	-3.509	-2.029	-0.628	1.883
	-0.349	1.300	-2.880	-1.195	-0.389	0.498	2.270
Tireage	1.393	0.922	-0.463	0.807	1.403	1.982	3.242
	0.995	0.743	-0.371	0.494	0.952	1.452	2.559
	0.403	0.414	-0.388	0.106	0.410	0.673	1.233
Interbelt	-0.604	0.431	-1.522	-0.873	-0.582	-0.319	0.147
	-0.608	0.402	-1.405	-0.878	-0.602	-0.336	0.155
	-0.456	0.303	-1.070	-0.642	-0.456	-0.247	0.112
Eb2b	-1.511	0.606	-2.916	-1.874	-1.464	-1.082	-0.471
	-1.402	0.552	-2.614	-1.740	-1.350	-1.026	-0.478
	-0.914	0.372	-1.687	-1.162	-0.905	-0.638	-0.234
Crnbblk	0.306	0.397	-0.438	0.038	0.291	0.579	1.105
	0.234	0.375	-0.463	-0.028	0.210	0.487	1.025
	0.108	0.297	-0.487	-0.083	0.106	0.324	0.678
λ_1	0.061	0.056	0.004	0.021	0.045	0.081	0.212
	0.072	0.069	0.008	0.028	0.050	0.094	0.243
	0.113	0.072	0.025	0.062	0.098	0.147	0.278
λ_2	1.889	1.576	0.169	0.734	1.478	2.572	5.955
	1.925	1.553	0.218	0.818	1.522	2.525	6.069
	2.471	1.514	0.492	1.361	2.162	3.162	6.391

45.5 Discussion

45.5.1 Alternative Prior Distributions

In addition to Zellner's g -priors, there are several other alternative proper priors to guarantee the posterior propriety under the monotone (partial) likelihood problem.

Jeffreys-Type Prior Recall that the posterior distribution for β is given by (45.11). Let $|\cdot|$ denote determinant. If we take

$$\pi(\beta) \propto |\mathbf{I}(\beta)|^{1/2} \quad (45.14)$$

in (45.11), the proposal by Heinze and Schemper [13] has a Bayesian interpretation under the Jeffreys-type prior for β . They also noted that the penalty function $|\mathbf{I}(\beta)|^{1/2}$ in the penalized likelihood function of the Cox proportional hazards model, which is $L_p^*(\beta|\mathcal{D}_{\text{obs}}) = L_p(\beta|\mathcal{D}_{\text{obs}})|\mathbf{I}(\beta)|^{1/2}$, is exactly the Jeffreys-type prior. If condition C1 in (45.13) holds, then $\pi(\beta)$ is proper, and thus the posterior distribution is proper because $L_p(\beta|\mathcal{D}_{\text{obs}})$ is bounded.

Shifted Jeffreys-Type Prior Let β_M be a mode of the prior distribution in (45.14). By adding β_M to β , we get a shifted Jeffreys-type prior given by $\pi_s(\beta) \propto |\mathbf{I}(\beta + \beta_M)|^{1/2}$, so that its mode is shifted to $\beta = \mathbf{0}$. Using $\pi_s(\beta)$, a different posterior $\pi_s(\beta|\mathcal{D}_{\text{obs}})$ is obtained from (45.11). The simulation study in [29] empirically suggests that the shifted Jeffreys-type prior may potentially reduce biases in MPLEs and posterior estimates of the regression coefficients.

Jeffreys-Type Prior Based on the First Risk Set Let $y_{(1)}, \dots, y_{(n)}$ denote the observed times arranged in ascending order associated with $\mathbf{x}_{(1)}, \dots, \mathbf{x}_{(n)}$ and $\nu_{(1)}, \dots, \nu_{(n)}$. Let $\pi_f(\beta)$ denote another variation of the Jeffreys-type prior, which only depends on the first risk set $\mathcal{R}(y_{i_0})$, where $i_0 = \min\{i \in \{1, \dots, n\} : \nu_{(i)} = 1\}$. Let $\mathbf{X}_{(i_0)}^*$ be the submatrix of \mathbf{X}^* corresponding to the first risk set. If $\mathbf{X}_{(i_0)}^*$ is of full column rank, then $\pi_f(\beta)$ exists and is proper. One benefit of this prior is the computing time. Using the first risk set to build the prior saves computation time especially for data sets with a large number of subjects and covariates.

Furthermore, constructing the prior based on the first risk set will not lose much information [29].

log-F Prior Almeida et al. [1] proposed the log-F($l_1/2, l_2/2$) as a solution for the monotone (partial) likelihood issue, where l_1 and l_2 are the degrees of freedom of the F distribution.

Different priors may result in different posterior summaries, interpretations, and degrees of difficulty in implementations, which deserves further research.

45.5.2 Bayesian Model Comparison

SAS and WinBUGS only produce DIC values for Bayesian model comparison. There are several other model selection criteria for Bayesian analysis, including the logarithm of the pseudomarginal likelihood (LPML) [15], the Watanabe-Akaike information criterion (WAIC) [28], and the Bayes factor [10], among others. However, it is still unknown whether a “better” prior may be selected under these criteria compared to DIC in the presence of monotone likelihoods.

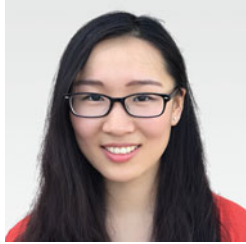
45.5.3 Software

The solution to the monotone likelihood problem proposed by Heinze and Schemper [13] is built on the idea of penalization investigated by Firth [8]. For recent accounts on the role of penalization for coping with the monotone likelihood problem, see [11] and [1]. SAS, SPLUS, and R codes for inference in the Cox proportional hazards model using the penalized partial likelihood function were implemented by Heinze and Ploner [12] and Ploner and Heinze [20]. The MPLE of β is a solution to the estimating equation $U(\beta) = \mathbf{0}$, where $U(\beta) = \partial \log L_p(\beta | \mathcal{D}_{\text{obs}}) / \partial \beta$ and $L_p(\beta | \mathcal{D}_{\text{obs}})$ is as in (45.6). In the approach by Heinze and Schemper [13], the estimate $\hat{\beta}^*$ is obtained by solving the equation $U^*(\beta) = \mathbf{0}$, where $U^*(\beta) = U(\beta) + \text{trace}\{I(\beta)^{-1} \partial I(\beta) / \partial \beta\} / 2$, with $I(\beta)$ denoting minus the Hessian matrix of $\log L_p(\beta | \mathcal{D}_{\text{obs}})$. The second term of $U^*(\beta)$ acts as a penalty. Bayesian approaches discussed in Sect. 45.3 can also be implemented using other softwares such as R [22], JAGS [21], and Stan [3].

References

- Almeida, F.M., Colosimo, E.A. Mayrink, V.D.: Prior specifications to handle the monotone likelihood problem in the Cox regression model. *Stat. Interf.* **11**, 687–698 (2018)
- Bryson, M.C., Johnson, M.E.: The incidence of monotone likelihood in the Cox model. *Technometrics* **23**, 381–383 (1981)
- Carpenter, B., Gelman, A., Hoffman, M.D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., Riddell, A.: Stan: A probabilistic programming language. *J. Stat. Softw.* **76**, 1–32 (2017)
- Chen, M.-H., Ibrahim, J.G., Shao, Q.-M.: Posterior propriety and computation for the Cox regression model with applications to missing covariates. *Biometrika* **93**, 791–807 (2006)
- Chen, M.-H., Ibrahim, J.G., Shao, Q.-M.: Maximum likelihood inference for the Cox regression model with applications to missing covariates. *J. Multivar. Anal.* **100**, 2018–2030 (2009)
- Cox, D.R.: Regression models and life-tables. *J. R. Stat. Soc. B* **34**, 187–220 (1972)
- Cox, D.R.: Partial likelihood. *Biometrika* **62**, 269–276 (1975)
- Firth, D.: Bias reduction of maximum likelihood estimates. *Biometrika* **80**, 27–38 (1993)
- Ge, M., Chen, M.-H.: Bayesian inference of the fully specified sub-distribution model for survival data with competing risks. *Lifetime Data Anal.* **18**, 339–363 (2012)
- Goodman, S.N.: Toward evidence-based medical statistics. 2: The Bayes factor. *Ann. Intern. Med.* **130**, 1005–1013 (1999)
- Greenland, S., Mansournia, M.A.: Penalization, bias reduction, and default priors in logistic and related categorical and survival regressions. *Stat. Med.* **34**, 3133–3143 (2015)
- Heinze, G., Ploner, M.: SAS and SPLUS programs to perform Cox regression without convergence problems. *Comput. Methods Programs Biomed.* **67**, 217–223 (2002)
- Heinze, G., Schemper, M.: A solution to the problem of monotone likelihood in Cox regression. *Biometrics* **57**, 114–119 (2001)
- Ibrahim, J.G., Chen, M.-H., Sinha, D.: *Bayesian Survival Analysis*. Springer, New York (2001)
- Ibrahim, J.G., Chen, M.-H., Sinha, D.: *Bayesian survival analysis*. Wiley StatsRef: Statistics Reference Online (2014). <https://doi.org/10.1002/9781118445112.stat06003>
- Kalbfleisch, J.D.: Non-parametric Bayesian analysis of survival time data. *J. R. Stat. Soc. B* **40**, 214–221 (1978)
- Kalbfleisch, J.D., Prentice, R.L.: *The Statistical Analysis of Failure Time Data*, 2nd edn. Wiley, Hoboken (2002)
- Krivtsov, V.V., Tananko, D.E., Davis, T.P.: Regression approach to tire reliability analysis. *Reliab. Eng. Syst. Safe.* **78**, 267–273 (2002)
- Lunn, D., Thomas, A., Best, N., Spiegelhalter, D.: WinBUGS—a Bayesian modelling framework: concepts, structure, and extensibility. *Stat. Comput.* **10**, 325–337 (2000)
- Ploner, M., Heinze, G.: *coxphf: Cox regression with Firth’s penalized likelihood* (2010). R package version 1.05 (<http://CRAN.R-project.org/package=coxphf>)
- Plummer, M.: JAGS Version 4.3.0 user manual (2017). <http://mcmc-jags.sourceforge.net>
- R Core Team.: *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria (2019)
- Roy, V., Hobert, J.P.: Convergence rates and asymptotic standard errors for Markov chain Monte Carlo algorithms for Bayesian probit regression. *J. R. Stat. Soc. B* **69**, 607–623 (2007)
- SAS Institute Inc.: *SAS/STAT®13.2 User’s Guide*. SAS Institute Inc., Cary, NC (2014)
- Sinha, D., Ibrahim, J.G., Chen, M.-H.: A Bayesian justification of Cox’s partial likelihood. *Biometrika* **90**, 629–641 (2003)
- Spiegelhalter, D.J., Best, N.G., Carlin, B.P., van der Linde, A.: Bayesian measures of model complexity and fit. *J. R. Stat. Soc. B* **64**, 583–639 (2002)
- Wang, W., Hu, C.: Proportional hazards regression models. In: Pham, H. (ed.) *Handbook of Engineering Statistics*, pp. 387–396, London. Springer (2006)
- Watanabe, S.: Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *J. Mach. Learn. Res.* **11**, 3571–3594 (2010)

29. Wu, J., de Castro, M., Schifano, E.D., Chen, M.-H.: Assessing covariate effects using Jeffreys-type prior in the Cox model in the presence of a monotone partial likelihood. *J. Stat. Theory Pract.* **12**, 23–41 (2018)
30. Zellner, A.: On assessing prior distributions and Bayesian regression analysis with g -prior distributions. In: Goel, P.K., Zellner, A. (eds.) *Bayesian Inference and Decision Techniques*, vol. 6 of *Studies in Bayesian Econometrics and Statistics*, pp. 233–243. North-Holland, Amsterdam (1986)



Jing Wu received her Ph.D. in statistics from the University of Connecticut in 2017. She works now at the University of Rhode Island, USA.



Mário de Castro received his Ph.D. from Universidade de São Paulo, Brazil in 2001. He worked at Universidade Federal do Espírito Santo, Brazil. In 2011, he was a visiting scholar at University of Connecticut, USA. He works now at Universidade de São Paulo in São Carlos, Brazil.



Ming-Hui Chen received his Ph.D. from Purdue University in 1993. He worked at Worcester Polytechnic Institute from 1993 to 2001. He has worked at University of Connecticut since fall 2001.



Multivariate Modeling with Copulas and Engineering Applications

46

Jun Yan

Contents

46.1	Introduction	932
46.2	Copulas and Multivariate Distributions	933
46.2.1	Copulas.....	933
46.2.2	Copulas to Multivariate Distributions.....	933
46.2.3	Concordance Measures.....	934
46.2.4	Fréchet-Hoeffding Bounds.....	934
46.2.5	Simulation.....	935
46.3	Some Commonly Used Copulas	935
46.3.1	Elliptical Copulas.....	935
46.3.2	Archimedean Copulas.....	937
46.4	Statistical Inference	939
46.4.1	Exact Maximum Likelihood.....	939
46.4.2	Inference Functions for Margins (IFM).....	939
46.4.3	Canonical Maximum Likelihood (CML).....	940
46.5	Engineering Applications	940
46.5.1	Multivariate Process Control.....	940
46.5.2	Degradation Analysis.....	942
46.6	Conclusion	944
References	944

Abstract

This chapter reviews multivariate modeling using copulas with illustrative applications in engineering such as multivariate process control and degradation analysis. A copula separates the dependence structure of a multivariate distribution from its marginal distributions. Properties and statistical inferences of copula-based multivariate models are discussed in detail. Applications in engineering are illustrated via examples of bivariate process control and degradation analysis, using existing data in the literature. An R package `copula` facilitates developments and applications of copula-based methods. The major change

from the last version (Yan 2006) is the update on the R package `copula` (Hofert et al. 2018).

Section 46.1 provides the background and motivation of multivariate modeling with copulas. Most multivariate statistical methods are based on the multivariate normal distribution, which cannot meet the practical needs to fit non-normal multivariate data. Copula-based multivariate distributions offer much more flexibility in modeling various non-normal data. They have been widely used in insurance, finance, risk management, and medical research. This chapter focuses on their applications in engineering.

Section 46.2 introduces the concept of copulas and its connection to multivariate distributions. The most important result about copulas is Sklar's (1959) theorem which shows that any continuous multivariate distribution has a canonical representation by a unique copula and all its marginal distributions. Scale-invariant dependence measures for two variables, such as Kendall's tau and Spearman's rho, are completely determined by their copula. The extremes of these two concordance measures, -1 and 1 , are obtained under perfect dependence, corresponding to the Fréchet-Hoeffding lower and upper bounds of copulas, respectively. A general algorithm to simulate random vectors from a copula is also presented.

Section 46.3 introduces two commonly used classes of copulas: elliptical copulas and Archimedean copulas. Elliptical copulas are copulas of elliptical distributions. Two most widely used elliptical copulas, the normal copula and the t copula, are discussed. Archimedean copulas are constructed without referring to distribution functions and random variables. Three popular Archimedean families, Clayton copula, Frank copula, and Gumbel copula, each having a mixture representation with a known frailty distribution, are discussed. Simulation algorithms are also presented.

Section 46.4 presents the maximum likelihood inference of copula-based multivariate distributions given the data. Three likelihood approaches are introduced.

J. Yan (✉)

Department of Statistics, University of Connecticut, Storrs, CT, USA
e-mail: jun.yan@uconn.edu

The exact maximum likelihood approach estimates the marginal and copula parameters simultaneously by maximizing the exact parametric likelihood. The inference functions for margins approach is a two-step approach, which estimates the marginal parameters separately for each margin in a first step and then estimates the copula parameters given the the marginal parameters. The canonical maximum likelihood approach is for copula parameters only, using uniform pseudo-observations obtained from transforming all the margins by their empirical distribution functions.

Section 46.5 presents two novel engineering applications. The first example is a bivariate process control problem, where the marginal normality seems appropriate, but joint normality is suspicious. A Clayton copula provides better fit to the data than a normal copula. Through simulation, the upper control limit of Hotelling's T^2 chart based on normality is shown to be misleading when the true copula is a Clayton copula. The second example is a degradation analysis, where all the margins are skewed and heavy-tailed. A multivariate gamma distribution with normal copula fits the data much better than a multivariate normal distribution.

Section 46.6 concludes and points to references about other aspects of copula-based multivariate modeling that are not discussed in this chapter.

Keywords

Association · Clustered data · Degradation · Failure time · Longitudinal data · Multivariate analysis · Process control

46.1 Introduction

Multivariate methods are needed wherever independence cannot be assumed among the variables under investigation. Multivariate data are encountered in real life much more often than univariate data. This is especially true nowadays with the rapid growth of data-acquisition technology. For example, a quality control engineer may have simultaneous surveillance of several related quality characteristics or process variables; a reliability analyst may measure the amount of degradation for a certain product repeatedly over time. Because of the dependence among the multiple quality characteristics and repeated measurements, univariate methods are invalid or inefficient. Multivariate methods that can account for the multivariate dependence are needed.

Classic multivariate statistical methods are based on the multivariate normal distribution. Under multivariate normality, an elegant set of multivariate techniques, such as principal

component analysis and factor analysis, has become standard tools and been successful in a variety of application fields. These methods have become so popular that often times they are applied without a careful check about the multivariate normal assumption. Non-normality can occur in different ways. First, the marginal distribution of some variables may not be normal. For instance, in the degradation analysis in Sect. 46.5, the error rates of magnetic-optic disks at all time points are skewed and heavy-tailed and hence cannot be adequately modeled by normal distributions. Second, even if all the marginal distributions are normal, jointly these variables may not be multivariate normal. For instance, in the bivariate process control problem in Sect. 46.5, marginal normality seems appropriate, but joint normality is suspicious. In both examples, multivariate distributions more flexible than the multivariate normal are needed.

Non-normal multivariate distributions constructed from copulas have proved very useful in recent years in many applications. A copula is a multivariate distribution function whose marginals are all uniforms over the unit interval. It is well-known that any continuous random variable can be transformed to a uniform random variable over the unit interval by its probability integral transformation. Therefore, a copula can be used to “couple” different margins together and construct new multivariate distributions. This method separates a multivariate distribution into two components, all the marginals and a copula, providing a very flexible framework in multivariate modeling. Comprehensive book references on this subject are Nelsen [39] and Joe [25]. For widely accessible introductions, see, for example, Genest and MacKay [17] and Fisher [11].

Copula-based models have gained much attention in various fields. Actuaries have used copulas in modeling dependent mortality and losses [13–15]. Financial and risk analysts have used copulas in asset allocation, credit scoring, default risk modeling, derivative pricing, and risk management [1, 3, 7]. Biostatisticians have used copulas in modeling correlated event times and competing risks [9, 49]. The aim of this chapter is to provide a review of multivariate modeling with copulas and to show that it can be extensively used in engineering applications.

The chapter is organized as follows. Section 46.2 presents the formal definition of copulas and construction of multivariate distribution with copulas. Section 46.3 presents details about two commonly used classes of copulas: elliptical copulas and Archimedean copulas. Section 46.4 presents likelihood-based statistical inferences for copula-based multivariate modeling. Section 46.5 presents two engineering applications, multivariate process control, and degradation analysis. Section 46.6 concludes and suggests future research directions. Implementations have been made easy by the open source software package `copula` [22] for the R project [41].

46.2 Copulas and Multivariate Distributions

46.2.1 Copulas

Consider a random vector $(U_1, \dots, U_p)^\top$, where each margin U_i , $i = 1, \dots, p$, is a uniform random variable over the unit interval. Suppose the joint cumulative distribution function (CDF) of $(U_1, \dots, U_p)^\top$ is

$$C(u_1, \dots, u_p) = \Pr(U_1 \leq u_1, \dots, U_p \leq u_p). \quad (46.1)$$

Then, function C is called a p -dimensional copula. As Embrechts et al. [7] noted, this definition of a copula masks some of the problems when constructing copulas using other techniques, by not explicitly specifying what properties a function must have to be a multivariate distribution function; for a more rigorous definition, see, for example, Nelsen [39]. However, it is “operational” and very intuitive. For example, one immediately obtains with this definition that for any p -dimensional copula C , $p \geq 3$, each $k \leq p$ margin of C is a k -dimensional copula and that independence leads to a product copula:

$$\Pi_p(u_1, \dots, u_p) = \prod_{i=1}^p u_i. \quad (46.2)$$

Every continuous multivariate distribution function defines a copula. Consider a continuous random vector $(X_1, \dots, X_p)^\top$ with joint CDF $F(x_1, \dots, x_p)$. Let F_i , $i = 1, \dots, p$, be the marginal CDF of X_i . Then, $U_i = F_i(X_i)$ is a uniform random variable over the unit interval. One can define a copula C as

$$C(u_1, \dots, u_p) = F\{F_1^{-1}(u_1), \dots, F_p^{-1}(u_p)\}. \quad (46.3)$$

The elliptical copulas in Sect. 46.3.1 are constructed this way. Another important class of copulas, Archimedean copulas, is constructed differently; see Sect. 46.3.2.

A copula (46.1) can be used to construct multivariate distributions with arbitrary margins. Suppose that it is desired that the i th margin X_i has marginal CDF G_i . A multivariate distribution function G can be defined via a copula C as

$$G(x_1, \dots, x_p) = C\{G_1(x_1), \dots, G_p(x_p)\}. \quad (46.4)$$

This multivariate distribution will have the desired marginal distributions.

Clearly, there is a close connection between copulas and multivariate distributions. It is natural to investigate the converse of (46.4). That is, for a given multivariate distribution function G , does there always exist a copula C such

that (46.4) holds? If so, is this C unique? These questions are answered by Sklar’s (1959) Theorem in the next subsection.

46.2.2 Copulas to Multivariate Distributions

Sklar’s Theorem is the most important result about copulas. The bivariate version of the theorem was established by Sklar [45] almost a half century ago in the probability metrics literature. The proof in the general p -dimensional case is more involved and can be found in Sklar [44]. A formal statement of the theorem is as follows [39, p.41].

Theorem 1 *Let F be a p -dimensional distribution function with margins F_1, \dots, F_p . Then there exists a p -dimensional copula C such that for all x in the domain of F ,*

$$F(x_1, \dots, x_p) = C\{F_1(x_1), \dots, F_p(x_p)\}. \quad (46.5)$$

If F_1, \dots, F_p are all continuous, the C is unique; otherwise, C is uniquely determined on $\text{Ran } F_1 \times \dots \times \text{Ran } F_p$, where $\text{Ran } H$ is the range of H . Conversely, if C is a p -dimensional copula and F_1, \dots, F_p are distribution functions, then the function F defined by (46.5) is a p -dimensional distribution function with marginal distributions F_1, \dots, F_p .

Sklar’s Theorem ensures that a continuous multivariate distribution can be separated into two components: univariate margins and multivariate dependence, where the dependence structure is represented by a copula. The dependence structure of a multivariate distribution can be analyzed separately from its margins. It is sufficient to study the dependence structure of a multivariate distribution by focusing on its copula.

The probability density function (PDF) of the CDF F in (46.5) can be found from the PDF of C and F_1, \dots, F_p . The PDF c of the copula C in (46.1) is

$$c(u_1, \dots, u_p) = \frac{\partial^p C(u_1, \dots, u_p)}{\partial u_1 \dots \partial u_p}. \quad (46.6)$$

When the density c is known, the density f of the multivariate distribution F in (46.5) is

$$f(x_1, \dots, x_p) = c\{F_1(x_1), \dots, F_p(x_p)\} \prod_{i=1}^p f_i(x_i), \quad (46.7)$$

where f_i is the density function of the distribution F_i . Expression (46.7) is called the canonical representation of a multivariate PDF, which can be used to construct likelihood for observed data.

46.2.3 Concordance Measures

The copula of two random variables completely determines any dependence measures that are scale-invariant, that is, measures remain unchanged under monotone increasing transformations of the random variables. The construction of the multivariate distribution (46.5) implies that the copula function C is invariant under monotone increasing transformations of its margins. Therefore, scale-invariant dependence measures can be expressed in terms of the copulas of the random variables.

Concordance measures of dependence are based on a form of dependence known as concordance. The most widely used concordance measures are Kendall's tau and Spearman's rho. Both of them can be defined by introducing a concordance function between two continuous random vectors (X_1, X_2) and (X'_1, X'_2) with possibly different joint distributions G and H but with common margins F_1 and F_2 . This concordance function Q is defined as

$$Q = \Pr\{(X_1 - X'_1)(X_2 - X'_2) > 0\} - \Pr\{(X_1 - X'_1)(X_2 - X'_2) < 0\}, \quad (46.8)$$

which is the difference between probability of concordance and dis-concordance of (X_1, X_2) and (X'_1, X'_2) . It can be shown that

$$Q = Q(C_G, C_H) = 4 \int_0^1 \int_0^1 C_G(u, v) dC_H(u, v) - 1, \quad (46.9)$$

where C_G and C_H are the copulas of G and H , respectively.

For a bivariate random vector (X_1, X_2) with copula C , Kendall's tau is defined as $Q(C, C)$, interpreted as the difference between the probability of concordance and dis-concordance of two independent and identically distributed observations. Therefore, we have

$$\tau = 4 \int_0^1 \int_0^1 C(u_1, u_2) dC(u_1, u_2) - 1, \quad (46.10)$$

where the range of τ can be shown to be $[-1, 1]$. Spearman's rho, on the other hand, is defined as $3Q(C, \Pi)$, where Π is the product copula obtained under independence. That is,

$$\rho = 12 \int_0^1 \int_0^1 u_1 u_2 dC(u_1, u_2) - 3. \quad (46.11)$$

The constant 3 scales this measure into the range of $[-1, 1]$ [see, e.g., 39, p.129]. Spearman's rho is proportional to the difference between the probability of concordance and dis-concordance of two vectors: both have the same margins, but one has copula C , while the other has the product copula Π . It is straightforward to show that Spearman's rho equals

to Pearson's product-moment correlation coefficient for the probability integral transformed variables $U_1 = F_1(X)$ and $U_2 = F_2(Y)$:

$$\begin{aligned} \rho &= 12E(U_1 U_2) - 3 = \frac{E(U_1 U_2) - 1/4}{1/12} \\ &= \frac{E(U_1 U_2) - E(U_1)E(U_2)}{\sqrt{\text{Var}(U_1) \text{Var}(U_2)}}. \end{aligned} \quad (46.12)$$

There are other dependence measures based on copulas. For example, tail dependence is a very important measure in studying the dependence between extreme events. Details can be found in Joe [25].

46.2.4 Fréchet-Hoeffding Bounds

Important bounds are defined for copulas and multivariate distributions. These bounds are called the Fréchet-Hoeffding Bounds, named after the pioneering work of Fréchet and Hoeffding, who independently published their work on this in 1935 and 1940, respectively [42]. Define functions M_p and W_p on $[0, 1]^p$ as follows:

$$\begin{aligned} M_p(u_1, \dots, u_p) &= \min(u_1, \dots, u_p), \\ W_p(u_1, \dots, u_p) &= \max(u_1 + \dots + u_p - n + 1, 0). \end{aligned}$$

Then for every copula C ,

$$W_p(u_1, \dots, u_p) \leq C(u_1, \dots, u_p) \leq M_p(u_1, \dots, u_p). \quad (46.13)$$

These bounds are general bounds, regardless of whether the margins are continuous or not. Function M_p is always a p -dimensional copula for $p \geq 2$. Function W_p is a copula only for $p = 2$; for $p \geq 3$, although not a copula, it is still the best possible lower bound since for any $u = (u_1, \dots, u_p) \in [0, 1]^p$, there exists a copula C (which depends on u) such that $C(u) = W_p(u)$. In the bivariate case, these bounds correspond to perfect negative dependence and perfect positive dependence, respectively. Within a given family of copulas, they may or may not be attained [see, e.g., 39, Table 4.1]. For $p = 2$, Fig. 46.1 shows the perspective plots of the Fréchet-Hoeffding bounds copulas and the product copula.

Intuitively, perfect dependence should lead to extremes of concordance measures. It can be shown that for continuous random vector (X_1, X_2) with copula C , $\tau = -1$ (or $\rho = -1$) is equivalent to $C = W_2$ and $\tau = 1$ (or $\rho = 1$) is equivalent to $C = M_2$; see [8] for a proof.

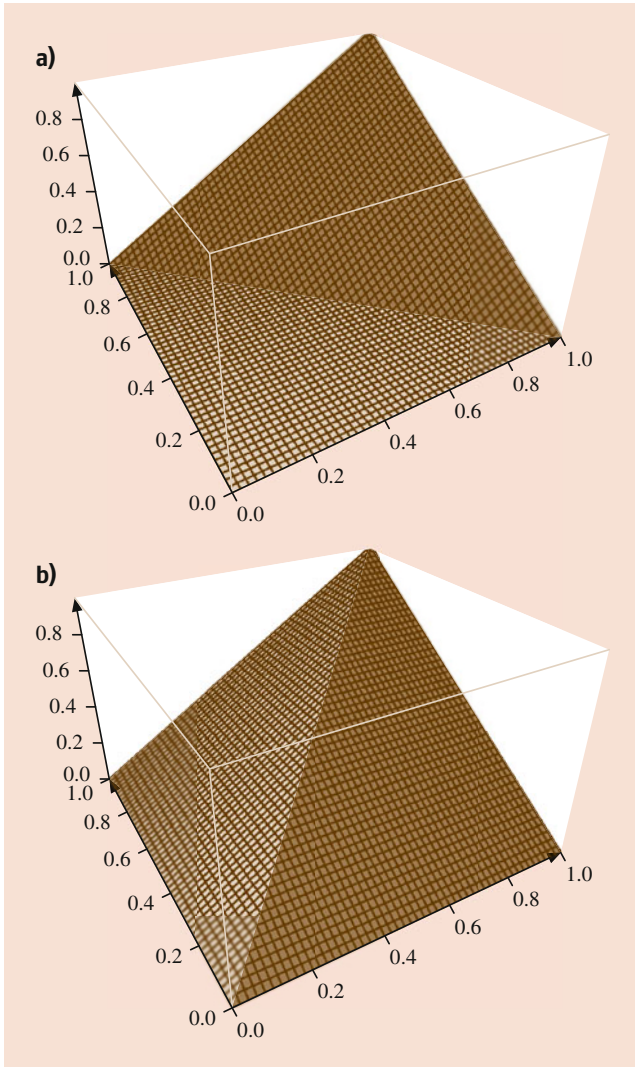


Fig. 46.1 Perspective plots of the Fréchet-Hoeffding bounds. (a) lower bound; (b) upper bound

46.2.5 Simulation

Random number generation from a copula is very important in statistical practice. Consider the p -dimensional copula in (46.1). Let $C_k(u_1, \dots, u_k) = C(u_1, \dots, u_k, 1, \dots, 1)$ for $k = 2, \dots, p - 1$. The conditional CDF of U_k given $U_1 = u_1, \dots, U_{k-1} = u_{k-1}$ is

$$C_k(u_k|u_1, \dots, u_{k-1}) = \frac{\partial^{k-1} C_k(u_1, \dots, u_k)}{\partial u_1 \dots \partial u_{k-1}} \cdot \frac{\partial^{k-1} C_{k-1}(u_1, \dots, u_{k-1})}{\partial u_1 \dots \partial u_{k-1}}. \quad (46.14)$$

Algorithm 1 is a general algorithm to generate a realization (u_1, \dots, u_p) from C via a sequence of conditioning. When the

Algorithm 1 Generating a random vector from a copula

1. Generate u_1 from a uniform over $[0, 1]$.
2. For $k = 2, \dots, p$, generate u_k from $C_k(\cdot|u_1, \dots, u_{k-1})$.

expression of $C_k(\cdot|u_1, \dots, u_{k-1})$ is available, a root finding routine is generally needed in generating u_k using the inverse CDF method. With realizations from C , one can easily generate realizations from the multivariate distribution (46.4) by applying the inverse CDF method at each margin.

46.3 Some Commonly Used Copulas

We introduce two commonly used copula classes in this section: elliptical copulas and Archimedean copulas. A third class of copulas, extreme value copulas, is very useful in multivariate extreme value theory but is omitted here to limit the scope of this chapter; more details about extreme value copulas can be found in Joe [25].

46.3.1 Elliptical Copulas

Elliptical copulas are copulas of elliptical distributions. A multivariate elliptical distribution of random vector (X_1, \dots, X_p) centered at zero has density of the form $\phi(t) = \psi(t^T \Sigma t)$, where $t \in R^p$ and Σ is a $p \times p$ dispersion matrix, which can be parameterized such that $\Sigma_{ij} = \text{Cov}(X_i, X_j)$ [10]. Let R_{ij} and τ_{ij} be Pearson's linear correlation coefficient and Kendall's tau between X_i and X_j , respectively. For an elliptical distribution, they are connected through

$$\tau_{ij} = \frac{2}{\pi} \arcsin(R_{ij}). \quad (46.15)$$

This relationship makes elliptical copulas very attractive in applications since the Kendall's tau matrix, similar to the correlation matrix, can offer a full range dependence structures. Tractable properties similar to those of multivariate normal make another attractiveness of elliptical copulas. The most popular elliptical distributions are multivariate normal and multivariate t , providing two popular copulas: normal copula and t copula.

The normal copula has been widely used in financial applications for its tractable calculus [1, 46]. Consider the joint CDF Φ_Σ of a multivariate normal distribution with correlation matrix Σ . Let Φ be the CDF of a standard normal variable. A normal copula with dispersion matrix Σ is defined as

$$C(u_1, \dots, u_p; \Sigma) = \Phi_\Sigma(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_p)). \quad (46.16)$$

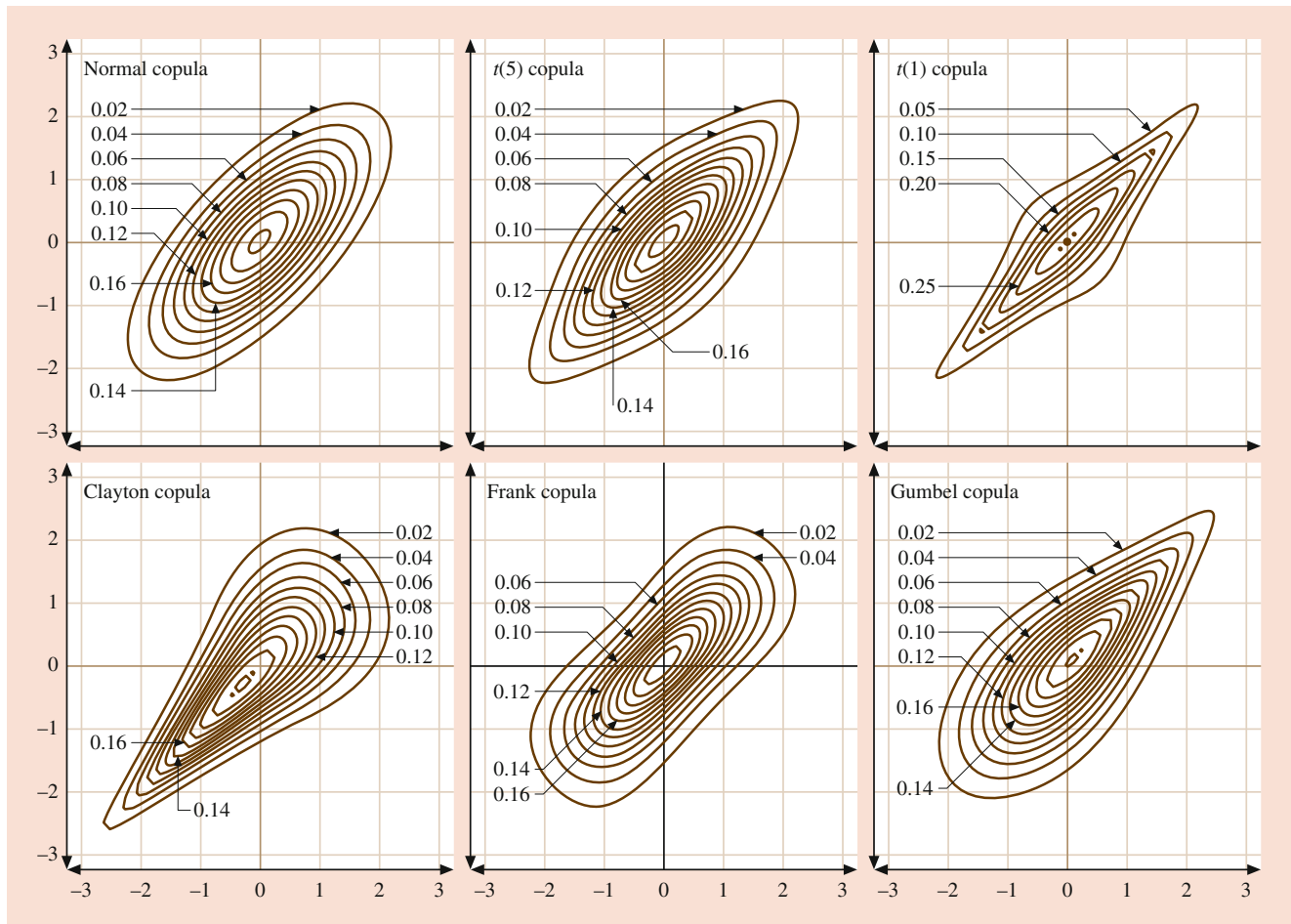


Fig. 46.2 Contours of bivariate distributions with the same marginals but different copulas. Both marginal distributions are standard normal

Functions Φ , Φ^{-1} , and Φ_{Σ} are available in any reasonably good statistical softwares, which makes their application widely accessible.

The t copula can be constructed similarly [6]. Consider the joint CDF $T_{\Sigma, \nu}$ of the standardized multivariate Student's t distribution with correlation matrix Σ and degrees of freedom ν . Let $F_{t_{\nu}}$ be the CDF of univariate t distribution with ν degrees of freedom. A t copula with dispersion matrix Σ and degrees of freedom parameter ν is defined as

$$C(u_1, \dots, u_p; \Sigma, \nu) = T_{\Sigma, \nu}(F_{t_{\nu}}^{-1}(u_1), \dots, F_{t_{\nu}}^{-1}(u_p)). \tag{46.17}$$

These copulas can be used to construct multivariate distributions using (46.5). Note that a normal copula with normal marginals is the same as a multivariate normal distribution. However, a t copula with t margins is not necessarily a multivariate t distribution. A multivariate t distribution must have the same degrees of freedom at all the margins. In contrast, a t copula with t margins can have different degrees

of freedom at different margins. It offers a lot more flexibility in modeling multivariate heavy-tailed data.

Figure 46.2 shows the density contours of bivariate distributions with same marginals but different copulas. These distributions all have standard normal as both margins, and their values of Kendall's tau are all 0.5. The three plots in the first row of Fig. 46.2 are for normal copula, t copula with 5 degrees of freedom, and t copula with 1 degree of freedom (or Cauchy copula). These densities are computed with (46.7). Note that a normal copula can be viewed as t copula with degrees of freedom being infinity. Figure 46.2 illustrates that the dependence at tails gets stronger as the degrees of freedom decrease.

Simulation from normal copulas and t copulas are straightforward if random number generators for multivariate normal and t are available. In R, package `mvtnorm` [18] provides CDF, PDF, and random number generation for multivariate normal and multivariate t distributions. These facilities are used in the implementation of package `copula` [22]. Figure 46.3 shows 1000 points from the corresponding bivariate distributions in Fig. 46.2.

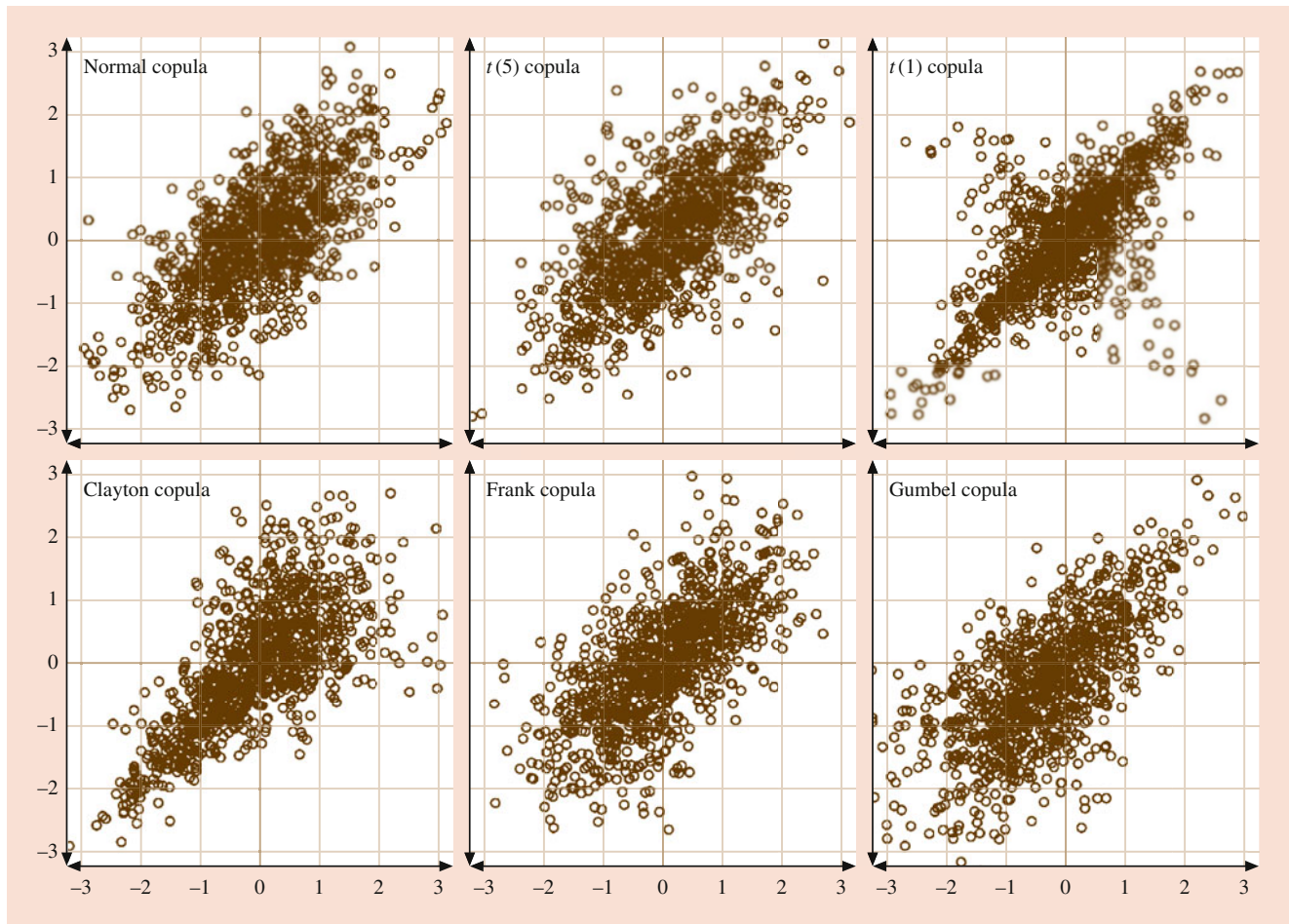


Fig. 46.3 1000 random points from bivariate distributions with the same marginals but different copulas. Both marginal distributions are standard normal

One caveat about elliptical copulas originates from the fact that not all elliptical distributions are marginally consistent [27]. Multivariate normal and t distributions have this property. For example, the d -dimensional marginal distribution for any p -dimensional normal distribution remains the same for all $p \geq d$. This property does not hold for some elliptical distributions such as the exponential power family, which has important practical consequences of the corresponding elliptical copulas [50]. See Example 3.1.9 of [22] for an illustration.

46.3.2 Archimedean Copulas

Archimedean copulas are constructed via a completely different route without referring to distribution functions or random variables. A key component in this way of construction is a complete monotonic function. A function $g(t)$ is completely monotonic on an interval J if it is continuous there and has derivatives of all orders which alternate in sign, that is,

$$(-1)^k \frac{d}{dt^k} \varphi(t) \geq 0, \quad k = 1, 2, \dots, \quad (46.18)$$

for all t in the interior of J . Let φ be a continuous strictly decreasing function from $[0, 1]$ to $[0, \infty]$ such that $\varphi(0) = \infty$ and $\varphi(1) = 0$, and let φ^{-1} be the inverse of φ . A function defined by

$$C(u_1, \dots, u_p) = \varphi^{-1} \{ \varphi(u_1) + \dots + \varphi(u_p) \} \quad (46.19)$$

is a p -dimensional copula for all $p \geq 2$ if and only if φ^{-1} is completely monotonic over $[0, \infty)$ [see, e.g., 39, Theorem 4.6.2]. The copula C in (46.19) is called an Archimedean copula. The name “Archimedean” for these copulas comes from a property of the unit cube and copula C which is an analog of the Archimedean axiom for positive real numbers [see 39, p.98 for more details]. The function φ is called the generator of the copula. A generator uniquely (up to a scalar multiple) determines an Archimedean copula.

In the bivariate case, an Archimedean copula may be obtained with weaker conditions on the generator φ and its pseudo-inverse $\varphi^{[-1]}$:

$$C(u_1, u_2) = \max[\varphi^{[-1]} \{\varphi(u_1) + \varphi(u_2)\}, 0], \quad (46.20)$$

where the generator φ is a function with two continuous derivatives such that $\varphi(1) = 0$, $\varphi'(u) < 0$, $\varphi''(u) > 0$ for all $u \in [0, 1]$, and $\varphi^{[-1]}$ is the pseudo-inverse of φ defined as

$$\varphi^{[-1]}(v) = \begin{cases} \varphi^{-1}(v) & 0 \leq v \leq \varphi(0), \\ 0 & \varphi(0) \leq v \leq \infty. \end{cases}$$

The generator φ is called a strict generator if $\varphi(0) = \infty$, in which case $\varphi^{[-1]} = \varphi$. [17] give proofs to some basic properties of bivariate copulas.

The generator φ plays an important role in the properties of an Archimedean copulas. It can be shown that the Kendall's tau for an Archimedean copula with generator φ is

$$\tau = 4 \int_0^1 \int_0^1 \frac{\varphi(v)}{\varphi'(v)} dv + 1. \quad (46.21)$$

This relationship can be used to construct estimating equations that equate the sample Kendall's tau to the theoretical value from the assumed parametric copula family.

Due to the exchangeable structure in (46.19), the associations among all variables are exchangeable too. As a consequence, an Archimedean copula cannot accommodate negative association unless $p = 2$. For Archimedean copulas with positive associations, there is a mixture representation due to Marshall and Olkin [34]. Suppose that conditional on a positive latent random variable called frailty, γ , the distribution of U_i is $F_i(U_i|\gamma) = U_i^\gamma$, $i = 1, \dots, p$, and U_1, \dots, U_p are independent. Then the copula C of U_1, \dots, U_p is

$$C(u_1, \dots, u_p) = E \left(\prod_{i=1}^p u_i^\gamma \right), \quad (46.22)$$

where the expectation is taken with respect to the distribution of γ , F_γ . Recall that the Laplace transform of γ is

$$\mathcal{L}(s) = E_\gamma(e^{-s\gamma}) = \int_0^\infty e^{-sx} dF_\gamma(x).$$

Laplace transform has well-defined inverse \mathcal{L}^{-1} . [34] show that the copula in (46.22) is

$$C(u_1, \dots, u_p) = \mathcal{L}(\mathcal{L}^{-1}(u_1) + \dots + \mathcal{L}^{-1}(u_p)). \quad (46.23)$$

This result suggests that an Archimedean copula can be constructed using the inverse of a Laplace transform as the generator.

Table 46.1 summarizes three commonly used one-parameter Archimedean copulas. A comprehensive list of one-parameter bivariate Archimedean copulas and their properties can be found in Table 4.1 of [39]. The three copulas in Table 46.1 all have inverse transforms of some positive random variables as their generators. The Clayton copula was introduced by Clayton [4] in modeling correlated survival times with a gamma frailty. The Frank copula first appeared in Frank [12]. It can be shown that the inverse of its generator is the Laplace transform of a log series random variables defined on positive integers. The Gumbel copula traces back to Gumbel [21]. Hougaard [24] uses a positive stable random variable to derive the multivariate distribution based on Gumbel copula.

Density contours of bivariate distributions constructed from these three Archimedean copulas are presented in the second row of Fig. 46.2. Both margins of these distributions are still standard normals. The parameters of these copulas are chosen such that the value of Kendall's tau is 0.5. The density of an Archimedean copula can be found by differentiating the copula as in (46.6). When the dimension p is high, the differentiation procedure can be tedious. Symbolic calculus softwares can be used for this purpose. From Fig. 46.2, one observes that the Frank copula has symmetric dependence. The dependence of the Clayton copula-based distribution is stronger in the lower-left region than in the upper-right region. In contrast, the dependence of the Gumbel copula-based distribution is stronger in the upper-right region than in the lower-left region.

Simulation from a general Archimedean can be done using the general Algorithm 1 in Sect. 46.2. When the inverse of the generator is known to be the Laplace transform of some positive random variable, an algorithm based on (46.23) is summarized in Algorithm 2 [14]. This algorithm is very easy to implement, given that a random number generator of the frailty is available. Gamma variable generator is available in most softwares. Algorithms for generating positive stable and log series variables can be found in Chambers et al. [2] and Kemp [28], respectively. For bivariate case, the general algorithm 1 can be simplified, avoiding numerical root finding. These algorithms have been implemented in package `copula` [22]. The lower panel of Fig. 46.3 shows 1000

Table 46.1 Some one-parameter (α) archimedean copulas

Family	Generator $\varphi(t)$	Frailty distribution	Laplace transformation of frailty $\mathcal{L}(s) = \varphi^{-1}(s)$
Clayton (1978)	$t^{-\alpha} - 1$	Gamma	$(1 + s)^{-1/\alpha}$
Frank (1979)	$\ln \frac{e^{\alpha t} - 1}{e^\alpha - 1}$	Log series	$\alpha^{-1} \ln(1 + e^s(e^\alpha - 1))$
Gumbel (1960)	$(-\ln t)^\alpha$	Positive stable	$\exp(-s^{1/\alpha})$

Algorithm 2 Generating a random vector from an Archimedean copula with a known frailty distribution

1. Generate a latent variable γ whose Laplace transformation \mathcal{L} is the inverse generator function φ^{-1} .
2. Generate independent uniform observations $v_1, \dots, v_p, i = 1, \dots, p$.
3. Output $u_i = \mathcal{L}(-\gamma^{-1} \log v_i), i = 1, \dots, p$.

random points generated from the corresponding bivariate distributions with Archimedean copulas in Fig. 46.2.

46.4 Statistical Inference

This section presents the maximum likelihood (ML) estimation for multivariate distributions constructed from copulas. Other methods, such as moment methods and nonparametric methods, are less developed for copula-based models and hence omitted.

Suppose that we observe a random sample of size n from a multivariate distribution (46.5):

$$(X_{i1}, \dots, X_{ip})^\top, \quad i = 1, \dots, n.$$

The parameter of interest is $\theta = (\beta^\top, \alpha^\top)^\top$, where β is the marginal parameter vector for the marginal distributions $F_i, i = 1, \dots, p$, and α is the association parameter vector for the copula C . Regression models for the marginal variables can be incorporated easily by assuming that the residuals follows a multivariate distribution (46.5).

46.4.1 Exact Maximum Likelihood

The exact log-likelihood $l(\theta)$ of the parameter vector θ can be expressed from (46.7):

$$l(\theta) = \sum_{i=1}^n \log c(F_1(X_{i1}; \beta), \dots, F_p(X_{ip}; \beta); \alpha) + \sum_{i=1}^n \sum_{j=1}^p \log f_j(X_{ij}; \beta). \tag{46.24}$$

The ML estimator of θ is

$$\hat{\theta}_{ML} = \arg \max_{\theta \in \Theta} l(\theta),$$

where Θ is the parameter space.

Under the usual regularity conditions for the asymptotic ML theory, the ML estimator $\hat{\theta}_{ML}$ is consistent and asymptotically efficient, with limiting distribution

$$\sqrt{n}(\hat{\theta}_{ML} - \theta_0) \rightarrow N(0, I^{-1}(\theta_0))$$

where θ_0 is the true parameter value and I is the Fisher information matrix. The asymptotic variance matrix $I^{-1}(\theta_0)$ can be estimated consistently by an empirical variance matrix of the influence functions evaluated at $\hat{\theta}_{ML}$.

In the `copula` package, a multivariate distribution constructed with a copula can be constructed with function `mvdc()`. Its log-likelihood can be computed with function `loglikMvdc()`. The maximum likelihood estimates and their standard errors are returned from function `fitMvdc()`.

The maximization of $l(\theta)$ in (46.24) may be a difficult task, especially when the dimension is high and/or the number of parameters is large. The separation of the margins and copula in (46.24) suggests that one may estimate the marginal parameters and association parameters in two steps, leading the method in the next subsection.

46.4.2 Inference Functions for Margins (IFM)

The estimation method of IFM was proposed by Joe and Xu [26]. This method estimates the marginal parameters β in a first step by

$$\hat{\beta} = \arg \max_{\beta} \sum_{i=1}^n \sum_{j=1}^p \log f_j(X_{ij}; \beta), \tag{46.25}$$

and then estimates the association parameters α given $\hat{\beta}$ by

$$\hat{\alpha} = \arg \max_{\alpha} \sum_{i=1}^n \log c(F_1(X_{i1}; \hat{\beta}), \dots, F_p(X_{ip}; \hat{\beta}); \alpha). \tag{46.26}$$

When each marginal distribution F_j has its own parameters β_j so that $\beta = (\beta_1^\top, \dots, \beta_p^\top)^\top$, the first step consists of an ML estimation for each margin $j = 1, \dots, p$:

$$\hat{\beta}_j = \arg \max_{\beta_j} \sum_{i=1}^n \log f_j(X_{ij}; \beta_j). \tag{46.27}$$

In this case, each maximization task has a very small number of parameters, greatly reducing the computational difficulty. This approach is called the two-stage parametric ML method by Shih and Louis [43] in a censored data setting.

The IFM estimator from (46.25) and (46.26), $\hat{\theta}_{IFM}$, is in general different from the ML estimate $\hat{\theta}_{ML}$. The limiting distribution of $\hat{\theta}_{IFM}$ is

$$\sqrt{n}(\hat{\theta}_{IFM} - \theta_0) \rightarrow N(0, G^{-1}(\theta_0))$$

where G is the Godambe information matrix [20]. This matrix has a sandwich form as from the usual robust estimation with estimating functions. Detailed expressions can be found in Joe [25]. Using pseudo-observations from the fitted probability integral transformation with $\hat{\beta}$ from (46.25), function `fitCopula()` called with `method = 'ml'` in the `copula` package returns point estimate of $\hat{\alpha}$ in (46.26). The returned standard error, however, should be ignored as the observations of the copula in the fitting are estimated instead of observed.

Compared to the ML estimator, the IFM estimator has advantages in numerical computations and is asymptotically efficient. Even in finite samples, it is highly efficient relative to the exact ML estimator [25]. The IFM estimate can be used as starting values in an exact ML estimation.

46.4.3 Canonical Maximum Likelihood (CML)

When the association is of explicit interest, parameter α can be estimated with the CML method without specifying the marginal distribution. This approach uses the empirical CDF of each marginal distribution to transform the observations $(X_{i1}, \dots, X_{ip})^\top$ into pseudo-observations with uniform margins $(U_{i1}, \dots, U_{ip})^\top$ and then estimates α as

$$\hat{\alpha}_{\text{CML}} = \arg \max_{\alpha} \sum_{i=1}^n \log c(U_{i1}, \dots, U_{ip}; \alpha). \quad (46.28)$$

The CML estimator $\hat{\alpha}_{\text{CML}}$ is consistent, asymptotically normal, and fully efficient at independence [16,43]. Function `fitCopula()` called with `method = 'mpl'`, which stands for maximum pseudo-likelihood, in the `copula` package returns point estimate $\hat{\alpha}_{\text{CML}}$ in (46.28) with valid standard errors.

46.5 Engineering Applications

Two engineering applications of copulas are considered in this section: multivariate process control and degradation analysis. An important third application is the modeling of multivariate failure times which may be censored. We focus on complete data applications in this chapter. In the example of multivariate process control, marginal normality seems appropriate, but joint normality is suspicious. In the example of degradation analysis, the margins are right-skewed and have long tails. We use a gamma distribution for each margin and a normal copula for the association.

46.5.1 Multivariate Process Control

In quality management, multiple process characteristics necessitate multivariate method for process control. There are three major control charts used in practice: Hotelling's T^2 , multivariate cumulative sum (MCUSUM), and multivariate exponentially weighted moving average (MEWMA); see Lowry and Montgomery [32] for a review. The most popular multivariate control chart is the T^2 chart, which has a long history since Hotelling [23]. Mason and Young [35] give details on how to use it with industrial applications. This method assumes that the multiple characteristics under surveillances are jointly normally distributed. The control limit of the chart is based on the sampling distribution of statistic T^2 , which can be shown to have an F distribution. When the multivariate normal assumption does not hold, due to either univariate or multivariate non-normality, T^2 control chart based on multivariate normality can be inaccurate and misleading.

Copula-based multivariate distributions open a new avenue for the statistical methods of multivariate process control. The parametric form of the multivariate distribution can be determined from a large amount of historical in-control data. Given a sample of observations when the process is in control, one can estimate the parameters and propose a statistic that measures the deviation from the target. The exact distribution of this statistic is generally unknown, and the control limit needs to be obtained from bootstrap; see for example Liu and Tang [31].

As an illustration, consider the example of bivariate process control in [33]. The data consists of 30 pairs of bivariate measurements from an exhaust manifold used on a Chrysler 5.21 engine in a given model year. They were collected from a machine capability study performed on the machine builder's floor. The sample correlation coefficient is 0.44. The left panel of Fig. 46.4 shows the scatter plot of the 30 observations. Normal assumption for each margin seems fine from the normal Q-Q plots (not shown). However, the joint distribution may not be a bivariate normal. The scatter plot suggests that the association may be stronger in the lower end than in the higher end of the data. This nonsymmetric association cannot be captured by a symmetric copula, such as those elliptical copula and Frank copula in Fig. 46.2. A better fit of the data may be obtained from a Clayton copula, which allows the bivariate dependence to be stronger at the left tail than at the right tail. The center panel of Fig. 46.4 shows the contours of the ML bivariate normal fit. The right panel of Fig. 46.4 shows the contours of the ML bivariate fit with normal margins and the Clayton copula. The maximized log-likelihood of the two models is 307.64 and 309.87, respectively. A formal test of the difference, which is beyond the scope of this chapter, can be done by comparing

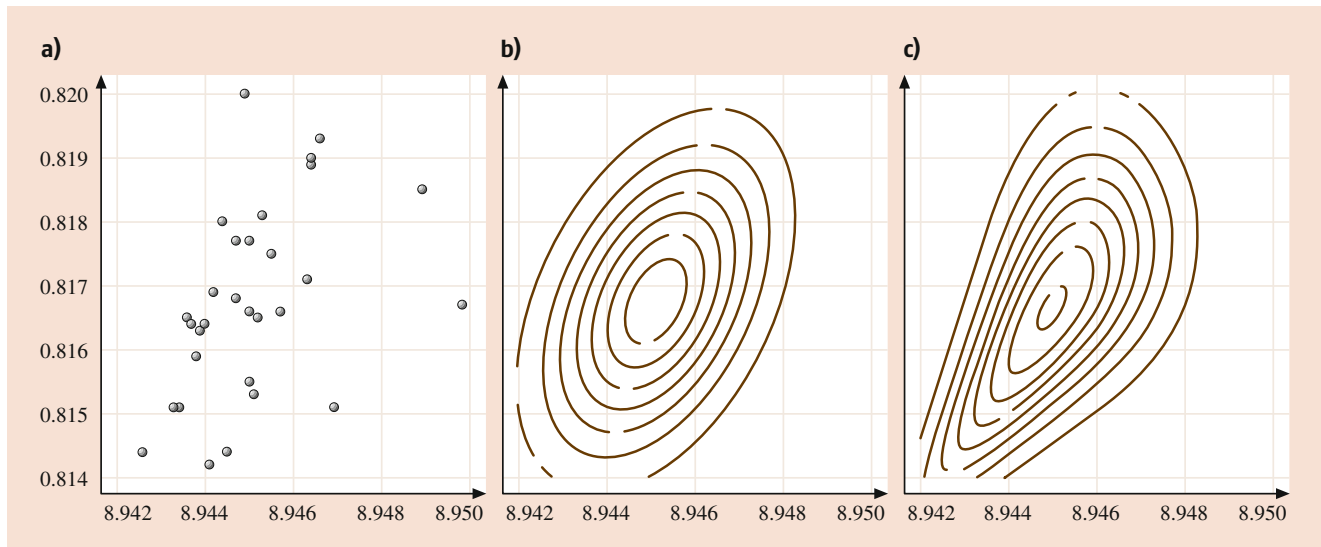


Fig. 46.4 Bivariate process characteristics and parametric fits. (a) scatter plot of the data; (b) contours of bivariate normal fit; (c) contours of bivariate fit with normal margins and Clayton copula

non-nested models without knowing the true model based on Kullback-Leibler information [48].

The T^2 control chart of Lu and Rudy [33] is a phase II chart for single observations to detect any departure of the underlying process from the standard values. Suppose that we observe a random sample of p -dimensional multivariate observations with sample size m . Let \bar{X}_m and S_m be the sample mean vector and sample covariance matrix, respectively. For a future p -dimension multivariate observation X , the T^2 is defined as

$$T^2 = (X - \bar{X}_m)^\top S_m^{-1} (X - \bar{X}_m). \quad (46.29)$$

Under joint normality, it can be shown that the exact distribution of

$$\frac{m^2 - mp}{p(m + 1)(m - 1)} T^2$$

is F with degrees of freedom p and $m - p$. The exact upper control limit for T^2 with level α is then

$$UCL_\alpha = \frac{p(m + 1)(m - 1)}{m^2 - mp} F_{1-\alpha; p, m-p}, \quad (46.30)$$

where $F_{1-\alpha; p, m-p}$ is the $100(1 - \alpha)$ percentile of an F distribution with p and $m - p$ degrees of freedom. In this example, $m = 30$, $p = 2$. The exact upper control limit for T^2 with level α is then

$$\begin{aligned} UCL_\alpha &= 2(30 + 1)(30 - 1)/[30^2 - 2(30)]F_{1-\alpha; 2, 28} \\ &= 2.14F_{1-\alpha; 2, 28}. \end{aligned}$$

With $\alpha = 0.9973$, the control limit $UCL = 15.75$.

When the true copula is a Clayton copula but is misspecified as a normal copula, the control limit in (46.30) can be inaccurate and hence misleading. By comparing the contours of a normal copula model with those of a Clayton copula model in Fig. 46.2, one can conjecture that if the true copula is a Clayton copula, and then $\Pr(T^2 > UCL_\alpha)$ will be greater than its nominal level α , because the bivariate density with Clayton copula is more concentrated on the lower-left part of the plot than the bivariate normal density. In other words, in order to maintain the control level α , one needs to increase the UCL of the T^2 chart. This difference obviously depends on the sample size m and the association parameter of the true Clayton copula. For a given sample size m and a Kendall's τ value, which determines the association strength of a Clayton copula, the control limit of T^2 can be obtained by simulation. Table 46.2 compares the 90%, 95%, 99%, and 99.73% percentiles of T^2 when the true copula is normal and when the true copula is Clayton. The percentiles under Clayton copulas are obtained from 100,000 simulations. The true Clayton copulas are parameterized to give Kendall's τ values 0.2, 0.4, 0.6, and 0.8. From Table 46.2, one observes that the simulated percentiles of T^2 are greater than those based on the F distribution under the normal assumption. The control region based on normal assumption is smaller than expected, which will result in investigating the process more often than necessary when the process is actually in control. The difference increases with the strength of the association.

This example illustrates that a non-normal joint distribution may have important influence on the control limit of the widely used T^2 chart, even when both the margins are normals. The T^2 statistic still measures the deviance from the target, but its distribution is unknown under the non-normal

Table 46.2 Comparison of T^2 percentiles when the true copula is normal and when the true copula is Clayton with various Kendall's τ . The percentiles under Clayton copulas are obtained from 100,000 simulations

Percentiles	Normal Copula	Clayton copula			
		$\tau = 0.2$	$\tau = 0.4$	$\tau = 0.6$	$\tau = 0.8$
90%	5.357	5.373	5.416	5.590	5.868
95%	7.150	7.253	7.468	8.061	9.396
99%	11.672	12.220	13.080	15.764	23.526
99.73%	15.754	16.821	18.611	24.173	41.123

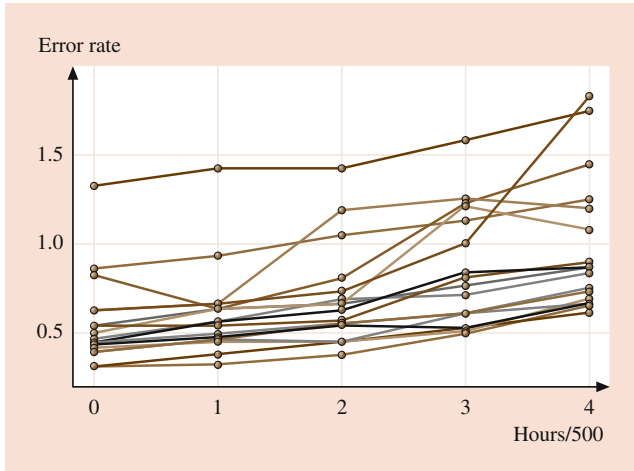


Fig. 46.5 Error rates ($\times 10^5$) of 16 magneto-optic data storage disks measured every 500 hours

model. A comprehensive investigation of multivariate process control using copula is a future research direction.

46.5.2 Degradation Analysis

Performance degradation data has repeated measures over time for each test unit [see, e.g., 36, chapter 13]. These repeated measures on the same unit are correlated. There is a voluminous statistical literature on analysis of repeated measurements [see, for example, 5]. Analysis of such data has been implemented in popular statistical softwares, for example, PROC MIXED of the SAS system [30] and the nlme package [40] for R and Splus. Continuous response variables are generally assumed to be normally distributed, and multivariate normal distribution is used in likelihood-based approaches. The following example shows that a multivariate gamma distribution with normal copula can provide much better fit to the data than a multivariate normal distribution.

Degradation data on block error rates of 16 magneto-optic data storage disks are collected every 500 hours for 2000 hours at 80°C and 85% relative humidity [38]. Figure 46.5 shows these error rates at all five time points. A degradation analysis often needs to fit a curve for the degradation trend

Table 46.3 IFM fit for all the margins using normal and gamma distributions, both parameterized by mean and standard deviation. Presented results are log-likelihood (Loglik), estimated mean, and estimated standard deviation (StdDev) for each margin under each model

Time in 500 h	Normal margins			Gamma margins		
	Loglik	Mean	StdDev	Loglik	Mean	StdDev
0	-0.484	0.565	0.062	2.568	0.565	0.054
1	-0.526	0.617	0.063	2.538	0.617	0.054
2	-2.271	0.709	0.070	-0.125	0.709	0.064
3	-4.441	0.870	0.080	-3.269	0.870	0.078
4	-6.996	1.012	0.094	-5.205	1.012	0.087

in order to do predictions at unobserved time points. Before choosing a curve to fit, we first do exploratory data analysis using the two-step IFM method to look into parametric modeling for each margin and for copula separately.

Separate parametric fit for each margin is the first step of the IFM approach in Sect. 46.4. Two parametric models for each margin are used: normal and gamma. To make the parameters comparable across models, the gamma distribution is parameterized by its mean μ and standard deviation σ , giving density function:

$$f(x; \mu, \sigma) = \frac{1}{\Gamma(\alpha)\beta^\alpha} x^{\alpha-1} e^{-\frac{x}{\beta}}, \quad (46.31)$$

where $\alpha = \mu^2/\sigma^2$ and $\beta = \sigma^2/\mu$. Table 46.3 summarizes the separate parametric fit for each margins using normal and gamma distributions. For all the margins, the gamma distribution fit yields higher log-likelihood than the normal distribution fit. The estimated mean from both models is the same for the first three digits after the decimal point. The estimated standard deviation is noticeably lower in the gamma model, especially at earlier time points where the data are more skewed and heavier tailed. These estimates are consistent with the descriptive statistics of each time point, suggesting that the mean error rate is increasing over time, and their standard errors is increasing with the mean level.

Given the parametric fit for each margins, we can explore copula fitting in the second step of IFM. Due to the small number of observations, we choose single parameter normal copulas with three dispersion structures: AR(1), exchangeable, and Toeplitz. In particular, with $p = 5$, the dispersion matrices with parameter ρ under these structures are, respectively,

$$\begin{pmatrix} 1 & \rho & \rho^2 & \rho^3 & \rho^4 \\ \rho & 1 & \rho & \rho^2 & \rho^3 \\ \rho^2 & \rho & 1 & \rho^2 & \rho^2 \\ \rho^3 & \rho^2 & \rho & 1 & \rho \\ \rho^4 & \rho^3 & \rho^2 & \rho & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & \rho & \rho & \rho & \rho \\ \rho & 1 & \rho & \rho & \rho \\ \rho & \rho & 1 & \rho & \rho \\ \rho & \rho & \rho & 1 & \rho \\ \rho & \rho & \rho & \rho & 1 \end{pmatrix}, \quad \text{and} \quad \begin{pmatrix} 1 & \rho & & & \\ \rho & 1 & \rho & & \\ & \rho & 1 & \rho & \\ & & \rho & 1 & \rho \\ & & & \rho & 1 \end{pmatrix}. \quad (46.32)$$

Table 46.4 IFM and CML fit for single parameter normal copulas with dispersion structures: AR(1), exchangeable, and Toeplitz

Dispersion Structure	IFM Fit				CML Fit	
	Normal margins		Gamma margins		Empirical margins	
	Loglik	$\hat{\rho}$	Loglik	$\hat{\rho}$	Loglik	$\hat{\rho}$
AR(1)	39.954	0.917	66.350	0.892	10.380	0.964
Exchangeable	38.618	0.868	62.627	0.791	9.791	0.942
Toeplitz	23.335	0.544	39.975	0.540	5.957	0.568

Table 46.4 summarizes the log-likelihood and the estimated association parameter ρ for the given estimated margins in Table 46.3. Note that the log-likelihood values are not comparable across models with different margins because the data being used in the estimation are different. They are comparable when the margins are modeled the same. For both normal margins and gamma margins, the AR(1) structure gives the highest log-likelihood value. The estimated parameter is about 0.9, indicating high dependence among repeated measurements.

Table 46.4 also presents the normal copulas estimation using the CML method. No parametric distribution is assumed for each margin. The empirical distribution is used to transform the observations of each margin into uniform variables in $[0, 1]$, which are then used in (46.28). The CML fit also shows that the AR(1) structure gives the highest log-likelihood and that the within disk dependence is high. Based on these exploratory analysis, the AR(1) structure is used for the dispersion matrix of normal copula in an exact ML analysis.

We now present the exact ML estimation of a degradation model. For the sake of simplicity, we use a linear function of time to model the mean $\mu(t)$ and a linear function of $\mu(t)$ to model the logarithm of the standard deviation $\sigma(t)$. That is,

$$\mu(t) = \phi_0 + \phi_1 t, \tag{46.33}$$

$$\log \sigma(t) = \psi_0 + \psi_1 [\mu(t) - 1.0]. \tag{46.34}$$

where $\phi_0, \phi_1, \psi_0,$ and ψ_1 are parameters and the function of $\log \sigma(t)$ is centered at 1.0 for easier predictions of the variance at higher error rates. Two parametric models are considered for the repeated error rates: (1) multivariate normal and (2) multivariate gamma via a normal copula. Note that the two models both uses the normal copula. The marginal distributions of the two models at time t are both parameterized by mean $\mu(t)$ and standard deviation $\sigma(t)$ for comparison purpose. Similar parameterization has been used in Lambert and Vandenhende [29] and Frees and Wang [15].

Table 46.5 summarizes the maximum likelihood estimate of the parameters and their standard errors for both models. These estimates for both marginal parameters and the copula parameter are virtually the same or very close to each other. However, the standard errors of these estimates are noticeably smaller in the multivariate gamma model. The maximized

Table 46.5 Maximum likelihood results to the disk error rate data. Parameter estimates, standard errors and log-likelihood are provided for both the multivariate normal model and the multivariate gamma model with a normal copula. The second entry of each cell is the corresponding standard error

Model	Marginal parameters				Copula Parameter	Loglik
	Mean		StdDev.			
	ϕ_0	ϕ_1	ψ_0	ψ_1	ρ	
Normal	0.564	0.099	-0.849	1.439	0.899	34.719
	0.057	0.019	0.262	0.557	0.034	
Gamma	0.564	0.101	-0.986	1.383	0.900	48.863
	0.051	0.015	0.185	0.442	0.033	

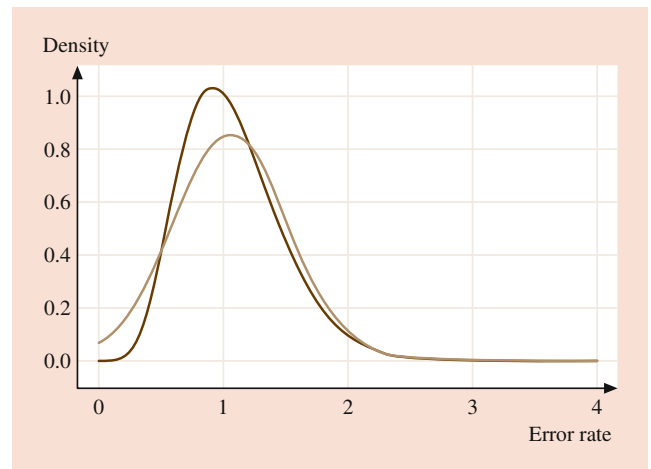


Fig. 46.6 Predictive density of disk error rate at 2500 h. The solid line is from the gamma model; the dashed line is from the normal model

log-likelihood from the gamma model is much higher than that from the normal model. Given that both models have the same number of parameters, the multivariate gamma distribution fits the data much better.

The difference between the two models can also be illustrated by their predictive density of the error rate at 2500 h. Figure 46.6 presents the densities of the error rate at 2500 h using the estimated mean $\mu(2500)$ and $\sigma(2500)$ obtained with $\hat{\phi}_0, \hat{\phi}_1, \hat{\psi}_0,$ and $\hat{\psi}_1$. The normal model gives mean 1.058 and standard deviation 0.465, while the gamma model gives mean 1.070 and standard deviation 0.411. Although the mean values are close, the gamma model gives small standard deviation. It captures the skewness and long tail of the data.

46.6 Conclusion

This chapter reviews multivariate modeling with copulas and provides novel applications in engineering. Multivariate distribution construction using copulas and their statistical inferences are discussed in detail. Engineering applications are illustrated via examples of bivariate process control and degradation analysis, using existing data in the literature. Copulas offer a flexible modeling strategy that separates the dependence structure from the marginal distributions. Multivariate distributions constructed with copula apply to a much wider range of multivariate scenarios than the traditionally assumed multivariate normal distribution. A publicly available R package `copula` [22] facilitates methods development applications.

Some important topics about copulas are not discussed in this chapter. Survival function is of great concern in failure time data analysis. Similar to (46.5), a multivariate survival function can be constructed from via a copula with

$$S(x_1, \dots, x_p) = C\{S_1(x_1), \dots, S_p(x_p)\},$$

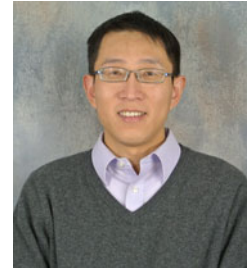
where S is the joint survival function and $S_i(t) = 1 - F(t)$ is the i th marginal survival function, $i = 1, \dots, p$. In this setting C is called a survival copula. Censoring presents an extra difficulty for multivariate failure time data analysis. Georges et al. [19] gives an excellent review on multivariate survival modeling with copulas. This chapter has focused on parametric copula models. Standard inferences of the maximum likelihood method can be applied under the usual regularity conditions. However, which copula to choose and how well it fits the data are important practical problems. Diagnostic tools, particularly graphical tools, can be very useful. See Hofert et al. [22] for a recent review.

Copulas have had a long history in the probability literature [42]. Recent development and application in insurance, finance, and biomedical research have been successful. With this chapter, it is hoped to encourage engineering researchers and practitioners to stimulate more advancement on copulas and seek more applications. Indeed, more engineering applications of copulas have appeared since the first version of this chapter [51], including multivariate process control [e.g., 47] and degradation analysis [e.g., 37], among others. Compared to other fields such as finance and hydrology, however, there seem to be potential to take fuller advantage of copulas in engineering applications.

References

1. Bouyè, E., Durrleman, V., Bikeghbali, A., Riboulet, G., Rconcalli, T.: Copulas for Finance – A Reading Guide and Some Applications. Working paper, Goupe de Recherche Opérationnelle, Crédit Lyonnais (2000)
2. Chambers, J.M., Mallows, C.L., Stuck, B.W.: A method for simulating stable random variables (corr: V82 P704; V83 P581). *J. Am. Stat. Assoc.* **71**, 340–344 (1976)
3. Cherubini, U., Luciano, E., Vecchiato, W.: *Copula Methods in Finance* Wiley, New York (2004)
4. Clayton, D.G.: A model for association in bivariate life tables and its application in epidemiological studies of familial tendency in chronic disease incidence. *Biometrika* **65**, 141–152 (1978)
5. Davis, C.S.: *Statistical Methods for the Analysis of Repeated Measurements*. Springer, Berlin (2002)
6. Demarta, S., McNeil, A.J.: The t copula and related copulas. *Int. Stat. Rev.* **73**, 111–129 (2005)
7. Embrechts, P., Lindskog, F., McNeil, A.: Modelling dependence with copulas and applications to risk management. In: Rachev, S. (ed.) *Handbook of Heavy Tailed Distribution in Finance*, pp. 329–384. Elsevier, Amsterdam (2003)
8. Embrechts, P., McNeil, A., Straumann, D.: Correlation and dependence in risk management: properties and pitfalls. In: Dempster, M. (ed.) *Risk Management: Value at Risk and Beyond*, pp. 176–223. Cambridge University Press, Cambridge (2002)
9. Escarela, G., Carrière, J.F.: Fitting competing risks with an assumed copula. *Stat. Methods Med. Res.* **12**, 333–349 (2003)
10. Fang, K.-T., Kotz, S., Ng, K.W.: *Symmetric Multivariate and Related Distributions*. Chapman & Hall, London (1990)
11. Fisher, N.I.: In: Kotz, S., Read, C.B., Banks, D.L. (eds.) *Encyclopedia of Statistical Sciences*, pp. 159–163. Wiley, London (1997)
12. Frank, M.J.: On the simultaneous associativity of $F(x, y)$ and $x + y - F(x, y)$. *Aequ. Math.* **19**, 194–226 (1979)
13. Frees, E.W., Carriere, J., Valdez, E.A.: Annuity valuation with dependent mortality. *J. Risk Insur.* **63**, 229–261 (1996)
14. Frees, E.W., Valdez, E.A.: Understanding relationships using copulas. *North Am. Actuarial J.* **2**, 1–25 (1998)
15. Frees, E.W., Wang, P.: Credibility using copulas. *North Am. Actuarial J.* **9**, 31–48 (2005)
16. Genest, C., Ghoudi, K., Rivest, L.-P.: A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika* **82**, 543–552 (1995)
17. Genest, C., MacKay, J.: The joy of copulas: bivariate distributions with uniform marginals (com: 87V41 P248). *Am. Stat.* **40**, 280–283 (1986)
18. Genz, A., Bretz, F., Hothorn, T.: *mvtnorm: Multivariate Normal and t distribution*. R package version 0.7-2
19. Georges, P., Lamy, A.-G., Nicolas, E., Quibel, G., Rconcalli, T.: *Multivariate Survival Modelling: A Unified Approach with Copulas*. Working paper, Goupe de Recherche Opérationnelle, Crédit Lyonnais (2001)
20. Godambe, V.P.: An optimum property of regular maximum likelihood estimation (ack: V32 p1343). *Ann. Math. Stat.* **31**, 1208–1212
21. Gumbel, E.J.: Bivariate exponential distributions. *J. Am. Stat. Assoc.* **55**, 698–707 (1960)
22. Hofert, M., Kojadinovic, I., Maechler, M., Yan, J.: *Elements of Copula Modeling with R*. Springer Use R! Series (2018)
23. Hotelling, H.: Multivariate quality control—illustrated by the air testing of sample bombsights. In: Eisenhart, C., Hastay, M.W., Wallis, W.A. (eds.) *Techniques of Statistical Analysis*, pp. 111–184. McGraw-Hill, New York (1947)
24. Hougaard, P.: A class of multivariate failure time distributions (corr: V75 p395). *Biometrika* **73**, 671–678 (1986)
25. Joe, H.: *Multivariate Models and Dependence Concepts*. Chapman, London (1997)
26. Joe, H., Xu, J.: *The Estimation Method of Inference Functions for Margins for Multivariate Models*. Technical Report 166, Department of Statistics, University of British Columbia (1996)
27. Kano, Y.: Consistency property of elliptical probability density functions. *J. Multivariate Anal.* **51**, 139–147 (1994)

28. Kemp, A.W.: Efficient generation of logarithmically distributed pseudo-random variables. *Appl. Stat.* **30**, 249–253 (1981)
29. Lambert, P., Vandenhende, F.: A copula-based model for multivariate non-normal longitudinal data: analysis of a dose titration safety study on a new antidepressant. *Stat. Med.* **21**, 3197–3217
30. Littell, R.C., Milliken, G.A., Stroup, W.W., Wolfinger, R.D.: *SAS System for Mixed Models*. SAS Institute (1996)
31. Liu, R.Y., Tang, J.: Control charts for dependent and independent measurements based on bootstrap methods. *J. Am. Stat. Assoc.* **91**, 1694–1700 (1996)
32. Lowry, C.A., Montgomery, D.C.: A review of multivariate control charts. *IIE Trans.* **27**, 800–810 (1995)
33. Lu, M.-W., Rudy, R.J.: Multivariate control chart. In: Pham, H. (ed.) *Recent Advances in Reliability and Quality Engineering*, pp. 61–74. World Scientific, Singapore (2001)
34. Marshall, A.W., Olkin, I.: Families of multivariate distributions. *J. Am. Stat. Assoc.* **83**, 834–841 (1988)
35. Mason, R.L., Young, J.C.: *Multivariate Statistical Process Control with Industrial Applications*. ASA-SIAM (2001)
36. Meeker, W.Q., Escobar, L.A.: *Statistical Methods for Reliability Data*. Wiley, London (1998)
37. Mireh, S., Khodadadi, A., Haghghi, F.: Copula-based reliability analysis of gamma degradation process and weibull failure time. *Int. J. Quality Reliab. Manag.* **36**, 654–668 (2019)
38. Murray, W.P.: Archival life expectancy of 3M magneto-optic media. *J. Magnet. Soc. Jpn.* **17**, 309–314 (1993)
39. Nelsen, R.B.: *An Introduction to Copulas*. Springer, Berlin (1999)
40. Pinheiro, J.C., Bates, D.M.: *Mixed-Effects Models in S and S-PLUS*. Springer, Berlin (2000)
41. R Core Team: *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna (2019)
42. Schweizer, B.: Thirty years of copulas. In: Dall’Aglia, G., Kotz, S., Salinetti, G. (eds.) *Advances in Probability Distributions with Given Margins: Beyond the Copulas*, pp. 13–50. Kluwer Academic Publishers, Dordrecht (1991)
43. Shih, J.H., Louis, T.A.: Inferences on the association parameter in copula models for bivariate survival data. *Biometrics* **51**, 1384–1399 (1995)
44. Sklar, A.: Random variables, distribution functions, and copulas—a personal look backward and forward. In: Rüschenendorf, L., Rüschenendorf, L., Schweizer, B., Taylor, M.D. (eds.) *Distributions with Fixed Marginals and Related Topics (IMS Lecture Notes Monograph Series, Volume 28)*, pp. 1–14. Institute of Mathematical Statistics (1996)
45. Sklar, A.W.: Fonctions de répartition à n dimension et leurs marges. *Publications de l’Institut de Statistique de l’Université de Paris* **8**, 229–231 (1959)
46. Song, P.X.-K.: Multivariate dispersion models generated from Gaussian copula. *Scand. J. Stat.* **27**, 305–320 (2000)
47. Verdier, G.: Application of copulas to multivariate control charts. *J. Stat. Plan. Inference* **143**, 2151–2159 (2013)
48. Vuong, Q.H.: Likelihood ratio tests for model selection and non-nested hypotheses (STMA V31 0456). *Econometrica* **57**, 307–333 (1989)
49. Wang, W., Wells, M.T.: Model selection and semiparametric inference for bivariate failure-time data (C/R: p73–76). *J. Am. Stat. Assoc.* **95**, 62–72 (2000)
50. Wang, X., Yan, J.: Practical notes on multivariate modeling based on elliptical copulas. *Journal de la Société Française de Statistique* **154**, 102–115 (2013)
51. Yan, J.: Multivariate modeling with copulas and engineering applications. In: Pham, H. (ed.) *Handbook of Engineering Statistics*, pp. 973–990. Springer, Berlin (2006)



Jun Yan received his PhD in Statistics from the University of Wisconsin - Madison in 2003. After four years in the Department of Statistics and Actuarial Science, he joined the Department of Statistics, University of Connecticut in 2007. His research interests are spatial extremes, survival analysis, multivariate dependence, and statistical computing, with applications in public health, environmental sciences, and sports.

Part VI

Applications in Engineering Statistics



Environmental Risks Analysis Using Satellite Data

47

Yuriy V. Kostyuchenko

Contents

47.1	Introduction	949
47.2	Method of Nonparametric Regularization of Multisource Data Distributions for the Analysis of Security and Risk Parameters	950
47.3	Multimodel Data Utilization in Risk Assessment Tasks: On the Method of Local Model Integration into Global Models	952
47.4	Approach to the Risk Assessment as the Analysis of Coherent Complex Measures of Multidimensional Multivariate Distributions	955
47.5	Notes on Decision-Making Approaches in Environmental Risk Analysis Using Satellite Data	957
47.6	Conclusions and Discussion	961
	References	961

Abstract

In this chapter the methodology of risk assessment and decision making in the field of environmental security is analyzed in view of complex novel threats and challenges connected with development of important multiscale tendencies of global climate and environmental changes, globalization, decentralization, and social transformation. To provide a methodological basis for increasing the effectiveness of environmental security management, a number of tasks were analyzed. A nonparametric two-stage method of multi-source data coupling and spatial–temporal regularization is proposed and discussed. Next, an approach to multiscale local and global model integration based on the modified

ensemble transform Kalman filtration procedure is proposed. An approach to a risk assessment based on the nonparametric kernel analysis of coherent complex measures of multidimensional multivariate distributions is then proposed. A decision making approach in the field of environmental risks analysis using satellite data and multi-model data is also considered and discussed. A number of important algorithms is described. Finally, the capabilities, limitations, and perspectives of the proposed methods and algorithms are discussed.

Keywords

Climate and environmental change · Multisource data · Observations and measurements · Data regularization · Nonlinear component analysis · Kernel copulas · Coherent risk measures · Model integration · Nonparametric risk analysis · Socio-environmental security

47.1 Introduction

The modern global community is facing complex novel threats and challenges connected with development of important multiscale tendencies of global climate and environmental changes, globalization, decentralization, and social transformation [1, 2]. Development of these tendencies not only generates new types of nexus, nonlinear interdependencies, and risks such as systems or chain risks, but also limits the applicability of traditional approaches to risk assessment [3, 4].

On the one hand, our understanding of multiscale and multiphysics catastrophic drivers is becoming more deep and comprehensive, which allows us to estimate a propagation of dangerous processes more accurately and correctly. as

Y. V. Kostyuchenko (✉)
Systemic Risk and Resilience Research Group, Advancing Systems Analysis Program, International Institute for Applied Systems Analysis (IIASA), Laxenburg, Austria
e-mail: kostyuchenko@iiasa.ac.at

Also, using modern GIS technologies and satellite observation leads to the increasing correctness of an estimation of the damage – spatial and temporal distribution of the infrastructure, houses, and other assets [4].

However, the number of resulting errors and, especially, uncertainties, is still high, and this substantially limits our ability to estimate risks. It is possible to say that the usual approach to analysis of risks in complex socio-environmental systems no longer leads to the correct assessment of losses, and requires application of significant fitness functions in every case. Studies show that the accuracy, correctness, reliability, and applicability of risk assessments vary widely depending on the locality.

This situation requires new tasks in the field of socio-environmental security to be formulated, and a new approach to operating with new comprehensive multisource data to be developed [5].

To provide a methodological basis for relatively reliable public safety from environmental dangers caused by natural or technological disasters, to support the state of human–environment dynamics, which includes restoration of the environment damaged by negative human behavior, to support the cycling of natural resources to products, to waste, to natural resources, to maintain the physical surroundings of society for its needs, aiming at the protection of society from social instability due to environmental degradation [5], it is necessary to propose a complex method to estimate the complex environmental risks and decision making, taking into account diverse multisource information.

Implementation of nonparametric methods for the analysis of complex multiscale, multicomponent, and multiphysics systems is widespread practice to obtain quantitative assessments of many important parameters, including risks [6]. In the context of novel threats, the key question is the methodology of application of these methods in view of the diversity of data sources.

Different data available have a different nature and require different approaches to be included in the decision-making process. Data from statistics, and from remote and direct observations and measurements should be combined, regularized, and also interpreted, verified, and calibrated using the wide range of local and global natural and data-assembling models, which should be harmonized (Fig. 47.1). This whole mass of data may be a basis for risk assessment and decision-making approaches in the field of socio-environmental security.

This methodology requires the development of interlinked approaches to multisource data analysis and regularization, to multiscale model integration, to multidriver risk assessment, and to decision making in complex multiphysics systems. In this chapter, such approaches will be considered.

47.2 Method of Nonparametric Regularization of Multisource Data Distributions for the Analysis of Security and Risk Parameters

The experience of study of the genesis, expansion, distribution of disasters, as well as the methods of assessment of the impact of the disasters to the socio-ecological systems, can lead to several important conclusions [7].

Analysis (including mapping) of the spatial and temporal distributions of varied types of disasters and data obtained from different sources is a very difficult task, and the direct comparison of these distributions is not entirely the correct method of analysis, first, because the long-term trends in changes in the distribution of certain types of drivers of disasters and their spatial and temporal scales obey different laws and do not coincide [8]. In most usual cases, for the assessment of socio-ecological system resilience, it is necessary to assess the overall danger from all types of natural (as well as from some types of technological) disaster. This requires the various driving forces, data from varied sources with different spatial and temporal scales, to be analyzed, and, thus, the different distributions of physical and economic parameters. This situation necessitates the determination of the measure of statistical distributions of available data (observations, measurements, archives, etc.), to the different spatial and temporal distributions of data.

Second, available data (statistics, archival records) usually have significant corrections, modifications, and errors. Thus, the data should be spatially and temporally regularized to reduce nonlinear uncertainties and errors.

To solve these problems – to obtain statistically reliable distributions of the disaster parameters per area unit during a certain period of time – a nonparametric data regularization algorithm may be proposed.

From the viewpoint of statistics, the proposed method may be presented as the following [8].

Correct statistical analysis requires the set of data x_i with controlled reliability, which reflects the distribution of investigated parameters over the study area during the whole observation period (taking into account variances in reliability of observation and archive data x_t). The set of observation data x_t ($x_t \in R^m$) consists of multisource data: historical records, archives, observations, measurements, etc., including data with sufficient reliability x_j ($x_j \in R^m$), where $j = 1, \dots, N$. The set x_t also includes observed and detected satellite indicators. The problem of the determination of controlled quality and reliability of the spatial–temporal distribution of the investigated parameters x_i might be solved within the framework of tasks of multivariate random processes analysis and multidimensional processes regularization [9].

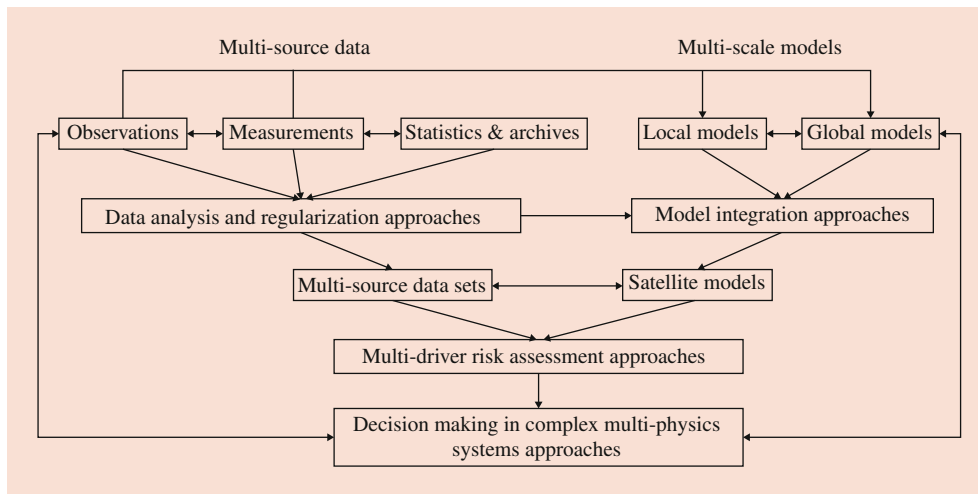


Fig. 47.1 Methodology of nonparametric methods implemented in tasks of socio-environmental security

Required regularization may be provided in different ways. If we are able to formulate a stable hypothesis on the distribution of the reliability of regional archives data within the framework of the defined problem we may be able to propose a relatively simple way of determining investigated parameters distributions $x_t^{(x,y)}$ toward distributions on measured sites x_t^m based on [10]:

$$x_t^{(x,y)} = \sum_{m=1}^n w_{x,y}(\tilde{x}_t^m) x_t^m, \quad (47.1)$$

where the weighting coefficients $w_{x,y}(\tilde{x}_t^m)$ are determined through the minimum task as:

$$\min \left\{ \sum_{m=1}^n \sum_{x_t^m \in R^m} w_{x,y}(\tilde{x}_t^m) \left(1 - \frac{x_t^m}{\tilde{x}_t^m} \right)^2 \right\}, \quad (47.2)$$

according to Cowpertwait [11]. Here, m is the number of records/points of measurements or observations; n is the number of observation series; x_t^m is the distribution of observation data; R^m is the set (aggregate collection) of observations; and \tilde{x}_t^m is the mean distribution of the measured parameters. This is the simple way of obtaining a regular spatial distribution of analyzed parameters over the study area, to which we can apply further analysis, in particular, temporal regularization.

A further regularization step should take into account both the observation distribution of temporal nonlinearity (caused by the imperfection of the available statistics) and the features of temporal–spatial heterogeneity of data distribution caused by the systemic complexity of the studied phenomena – natural and technological disasters. According to Mudelsee et al., Lee et al., and Villez et al. [12–14], the kernel-based

nonlinear approaches are quite effective for the analysis of such types of distributions.

Let us formulate the following task: using the method of data regularization, reduce the data of observations, measurements, and archive records received at random points, at random times, with uncertain accuracy regarding the true (most probable) state of the studied system.

The proposed two-step nonparametric regularization method is based on modified kernel principal component analysis (KPCA) [15–17].

First, it is necessary to find a set X , which closely corresponds to all available data distributions. If there are no additional assumptions about X , this task may be presented as the distribution of data in the discrete space H using the mapping function $\Phi: X \rightarrow H, x \mapsto \Phi(x)$. In particular, on the Hilbert space, this function will be presented as $\Phi: X \rightarrow \mathfrak{R}^X, x \mapsto K(\cdot, x)$. The correspondence of the resulting distribution of the input data will be determined by the point product of this space, which can be defined using the appropriate function: $K: X \times X \rightarrow \mathfrak{R}, (x, x') \mapsto K(x, x')$, called the kernel function, and all $x, x' \in X, K(x, x') = \langle \Phi(x), \Phi(x') \rangle_H$.

In the most usual cases the multisource data may be presented as the Gaussian random process with distribution $f: X \rightarrow \mathfrak{R}$, with mean $m: X \rightarrow \mathfrak{R}$, and with the covariation function $\mathbf{K}: X \times X \rightarrow \mathfrak{R}$. In this case $m(\mathbf{x}) = E[f(\mathbf{x})]$, and $\mathbf{K}(\mathbf{x}, \mathbf{x}') = E[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]$. A covariation matrix will determine a distribution of the vector of output, regularized data $f_x = (f(x_1), \dots, f(x_n))^T \approx N(\mathbf{0}, \mathbf{K}_{xx})$, which is the multivariate Gaussian distribution. In this case, covariation will define a form of the data distribution; thus, for the Gaussian processes a covariation will play the role of the kernel function.

Therefore, the task of regularization may be presented through the covariation matrix C as:

$$\min_{f \in H} \frac{1}{2} \|f\|_H^2 + C \sum_{i=1}^n \max \{0, \max (\Delta (x'_i, x') - (f(x_i, x'_i) - f(x_i, x')))\}, \tag{47.3}$$

where $f(\cdot, \cdot) = \sum_{i=1}^N \sum_{x \in X} \alpha_{ix} K((\cdot, \cdot), (x_i, x'))$.

If principal components algorithms (PCAs) will be used, the task of solving the system of equations:

$$C v_k = \lambda_k v_k, \tag{47.4}$$

should be formulated. The decomposition of eigenvalues of the empirical matrix of covariance:

$$C = E[(x - E(x))(x - E(x))^T], \tag{47.5}$$

should be executed. In the general case of Gaussian processes this solution may be presented as:

$$C v_k = \frac{1}{n} \sum_{i=1}^n \Phi(x_i) \Phi(x_i)^T v_k = \lambda_k v_k, \tag{47.6}$$

where $v_k = \sum_{i=1}^n \alpha_{ki} \Phi(x)$, $K\alpha = \lambda\alpha$, and values x for any principal component k may be calculated as:

$$(v_k, \Phi(x)) = \sum_{i=1}^n \alpha_{ki} K(x, x_i). \tag{47.7}$$

Thus, within the framework of this approach the algorithm of nonlinear regularization might be described as the following rule:

$$x_i = \sum_{i=1}^N \alpha_i^k \tilde{k}_t(x_i, x_i). \tag{47.8}$$

In this equation, the coefficients α selected according to the optimal balance of the relative validation function and the covariance matrix, for example, as [13]:

$$C^F v = \frac{1}{N} \sum_{j=1}^N \Phi(x_j) \Phi(x_j)^T \cdot \sum_{i=1}^N \alpha_i \Phi(x_i), \tag{47.9}$$

where the nonlinear mapping function of input data distribution Φ is determined as [15]:

$$\sum_{k=1}^N \Phi(x_k) = 0, \tag{47.10}$$

and \tilde{k}_t indicates the mean values of the kernel matrix $\mathbf{K} \in R^N$ ($[\mathbf{K}]_{ij} = [k(x_i, x_j)]$). The vector components of the matrix are determined as $\mathbf{k}_t \in R^N$; $[\mathbf{k}_t]_j = [k_t(\mathbf{x}_t, \mathbf{x}_j)]$. The matrix is calculated according to the modified rule of Cowpertwait and Mudelsee et al. [11, 12] as:

$$\mathbf{k}_t(\mathbf{x}_i, \mathbf{x}_t) = \left\langle \rho_{j,t}^{x_j} (1 - \rho_{j,i})^{x_j} \right\rangle, \tag{47.11}$$

where ρ indicates empirical parameters, selected according to the classification model of study phenomena [14].

Using the described algorithm it is possible to obtain regularized spatial-temporal distribution of the investigated parameters over the whole observation period with rectified reliability [12]. These distributions of multisource data should be used for the further modeling of risks.

It may be noted that the obtained regularized distributions can also be interpreted as the average for the observation interval probability of disaster occurrence in a certain area of the study territory.

47.3 Multimodel Data Utilization in Risk Assessment Tasks: On the Method of Local Model Integration into Global Models

The task of refinement of large-scale scenarios based on global models using local modeling should be considered. This task is fundamentally different from so-called regionalization of climatic, ecological, or runoff models. The task of the regionalization of models in the traditional sense is usually reduced to the “downscaling” tasks, or to the change (usually a decrease) in the spatial and temporal dimensions of the distributions obtained from the models. The proposed approach includes refinement of calculations based on the use of additional data sets not included in the standard model (typically, global models include 5–10% of the available measurements in the research area, which is obviously not sufficient for the calculation of local forecasts).

The principal complexity of this problem lies in the fact that climatic model parameters are usually calculated in the so-called geo-space, or, by definition, in the quasi-two-dimensional space of the earth’s surface characterized by closure, positive curvature (convexity), quasi-fractality, and anisotropy in geophysical fields (for example, in the gravitational, magnetic field of the Earth). Geospatial location is determined by spherical (latitude and longitude) or rectangular coordinates. The anisotropy of the geo-space is expressed in the unevenness of the horizontal and vertical directions (vertical movement is fairly complicated), and results in the sphericity of the Earth’s structure, as well as latitudinal

and longitudinal directions, the manifestation of which is latitudinal zonation. This determines the specific topology of data allocations for model calculations. In this situation, the results of observations are the sets of data obtained at certain points, and thus, although they are geo-referred, they do not necessarily correspond to all basic postulates that follow from the definition of geo-space.

From a mathematical viewpoint, the problem is to include the observation data (a priori stochastic data, which must be described by completely different methods and which are not necessarily complete) into the set of calculation data with the given topology (into the model data calculated on geo-space). In the general case, solving this task should solve the topological problem of bringing the topology of one set (with a given topology) with the number of sets of random data with an a priori indefinite topology.

In our case, the topology of the set of model data is given by the definition of geo-space, although the topology of sets of observational data is a priori not defined [18]. Based on the general theory of obtaining and analyzing data from experiments [9], we can analyze the data of current meteorological measurements in terms of the theory of Hilbert spaces, in which limited and unrestricted operators, and random elements and operators are defined, and thus the elements of linear and convex programming are determined. Thus, in our case, the use of the theory of finite-dimensional Hilbert spaces for the analysis of meteorological measurement data distributions may allow us to obtain results that will enable us to determine the topology of the corresponding sets and integrate data into sets with the topology, defined in the geo-space.

The representation of observations in Hilbert spaces allows us to stay within the framework of the so-called local simulation and thus avoid the need to determine boundary conditions dependent on the data structure [19]. In the future, this frees us from the need to harmonize the boundary conditions of two different types of data. Thus, the mathematical problem is reduced to finding the correct method for integrating meteorological observations, in the general case presented in the Hilbert random spaces, and simulation data presented as uniform normalized sets of data defined in the geo-space with the given topology.

Figuratively, this task can be represented as the gluing of patches of arbitrary shape and size on the surface of a round ball. In this case, the patches must be adjusted to each other, the total area of the patches is much smaller than area of the ball, the patches can have any shape and topology, and the ball has a finite elasticity, i.e., it can never change the positive curvature. The method proposed in this section is that the patches are pre-smoothed, their edges are cut and fitted to each other, after which they are consistently glued to the surface of the ball.

Thus, we propose to embed into the model the additional data not used during the construction of the model, and recalculate the necessary parameters on a grid that meets the needs of local simulation and is based on available data. This means that, unlike traditional downscaling, the proposed approach is based not on the results of simulation and on the properties of the resulting sets (with a substantially limited spatial and temporal resolution), but on the data of regional measurements. Therefore, the local data become decisive for the regionalization of the model in this approach, which is a fundamental difference.

Thus, the problem is divided into the problem of reducing the method of observing analysis and obtaining data measurement distributions, and the task of integrating the obtained reduced data into the model distributions with subsequent calculation of parameters on a reduced grid.

When we solve the problem of specifying the model on the sets of observational data presented as the Hilbert spaces in part of the reduction of the method of observation analysis, we will consider sets of data $\xi(x_{ij})$, measured within the framework of the method presented by linear operators $A_0 \rightarrow A$. To refine A , with sufficiently known (known and/or controlled epistemic reliability) signal $f'(x_{ij})$, represented by a random vector with a known covariance operator $F \in (\mathfrak{R} \rightarrow \mathfrak{R})$, $Fx = Ef(x, f)$, $x \in \mathfrak{R}$, we will be measuring the sets of variables [9]:

$$\xi' = Af' + v', \quad (47.12)$$

where v is a random element of a Hilbert space $\tilde{\mathfrak{R}}$ with a correlation operator $\Sigma x = Ev(x, v)$, $x \in \tilde{\mathfrak{R}}$, $Ev = 0$, which determines the error, or a “noise” – the measure of aleatoric uncertainty of measurements; ξ' indicates measurements refined with the model ξ ; E indicates mathematical expectation; A indicates the methodology, or the “model of measurements” are random linear operators (such that, $A \in B\mathfrak{R} \rightarrow \tilde{\mathfrak{R}}$, $x \in \mathfrak{R}$, $y \in \mathfrak{R}$, whereas the function $f(Ax, y) = f(x, A^*y)$) defines the method of data transformation.

If we introduce any measurable set M ($M \subset \tilde{\mathfrak{R}} \text{ B } \mathfrak{R}$), on which we define a random vector η , that describes the distribution of regularized data with a certain reliability, then in relation to it the problem of determining the methodology for integrating observation data into a general model, that is, refinement (reduction) operators, can be defined as [20]:

$$A_\eta = E(A|\eta), J_\eta = \left(E((A - A_\eta) * F(A - A_\eta) * |\eta) \right). \quad (47.13)$$

Here, F is a covariance data operator; J is the operator of uncertainty.

In this case, the task of reducing the data distributions can be formulated as:

$$\begin{aligned} E \left(\inf \left\{ E \left(\|R'\xi - Uf\|^2 | \eta \right) R' \in \left(\mathfrak{N} \rightarrow U \right) \right\} \right) \\ = E \left(E \left(\|R_\eta \xi - Uf\|^2 | \eta \right) \right) = E \|R_\eta \xi - Uf\|^2, \end{aligned} \tag{47.14}$$

where U is the orthogonal projector to the subspace \mathfrak{N} . Thus, we are talking about the determination of a random operator $R' = R_\eta$, which corresponds to the condition (47.20):

$$\begin{aligned} \inf \left\{ E \left(\|R'\xi - Uf\|^2 | \eta \right) R' \in \left(\mathfrak{N} \rightarrow U \right) \right\} \\ = E \left(\|R_\eta \xi - Uf\|^2 | \eta \right). \end{aligned} \tag{47.15}$$

After solving this problem, we obtain mutually coherent distributions of observation data with coordinated boundaries, based on the results of the analysis of data from individual measurements. The resulting sets, represented on Hilbert spaces, can be integrated (reduced) into the global models of any complexity with current topological properties.

Thus, at the first stage, the problem is reduced to the determination of regularized distributions of observation data. In addition, it should be noted that the problem of estimating the uncertainties of the distributions obtained is still urgent.

The task of obtaining the sets of statistically reliable distributions that are regularized in space and time of required indicators from the observed data (for example, from meteorological stations) in the research area can be solved using the proposed [14] algorithm for nonlinear spatial–temporal data regularization based on the analysis of the main components with the modified method of the smoothing nonlinear kernel function, kernel principal component analysis (KPCA) [8]. Using the proposed algorithm we obtain a regularized spatial–temporal distribution of the characteristics of the investigated parameters throughout the observation period with smooth reliability, taking into account all sources of observation [8, 14].

In accordance with the above general approach, the task is to integrate a plurality of data presented in the form of Hilbert spaces with given linear transformation operators to a set of data with a given topology (a set of model decisions).

Thus, based on the results of data analysis within the framework of the described method, we obtain a set of normalized distributions $\xi_t = A_t f(x_t) + v_t$, where t is the time (defined as the modeling step for a plurality of model data and as a measure of the data set). In the future, it can be proposed to jointly analyze the modeling and observation data in the modified ensemble transform Kalman filter (ETKF) modification method [21].

In this case, we assume that the vectors of the true state of the system x at time k are determined in accordance with the general law:

$$x_t = F_t x_{t-1} + B_t u_t + w_t, \tag{47.16}$$

where F_t is the matrix of the evolution of the system, or the simulated effects on the vector x_{t-1} at the moment $t - 1$; B_t is the matrix of controls, measured by the effects u_t on vector x ; w_t is a random process with the covariant matrix Q_t . In this way, we introduce the description of the model distributions F_x and the observation data B_t .

Let us determine the extrapolation value of the vector of the true state of the system by evaluating the state vector in the previous step:

$$\hat{x}_{t|t-1} = F_t \hat{x}_{t-1|t-1} + B_t u_{t-1}. \tag{47.17}$$

For this extrapolation value of the vector of the true state one can determine the general form of the covariance matrix:

$$P_{t|t-1} = F_t P_{t-1|t-1} F_t^T + Q_{t-1}. \tag{47.18}$$

The difference between the estimated (extrapolation) value of the vector of the true state of the system and that obtained at the appropriate simulation step can be estimated as:

$$\Delta \hat{x}_t = \xi_t - A_t \hat{x}_{t|t-1}, \tag{47.19}$$

a covariance matrix of deviation:

$$S_t = A_t P_{t|t-1} A_t^T + R_t. \tag{47.20}$$

Then, based on the covariance matrices of extrapolation to the state vector and the measurements, we introduce a matrix of optimal coefficients of the Kalman amplification:

$$K_t = P_{t|t-1} A_t^T S_t^{-1}. \tag{47.21}$$

Using this, we will adjust the extrapolation values of the vector of the true state of the system:

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t \Delta \hat{x}_t. \tag{47.22}$$

Also, we introduce a geo-referred filter for the distribution of the vector of the state x_{ij} , which will depend on geographically bound j , i and in the general case, this will not depend on the time t :

$$(x_{ij})_t = (x_{ij})_t^\alpha = (x_{ij})_t \alpha_{ij}. \tag{47.23}$$

Here, the coefficients α are chosen according to the introduced KPCA algorithm [13], according to the rule of estimat-

ing the optimal balance of the mutual validation function and the covariance matrix:

$$C^F v = \frac{1}{N} \sum_{j=1}^N \Phi(x_j) \Phi(x_j)^T \cdot \sum_{i=1}^N \alpha_i \Phi(x_i), \quad (47.24)$$

where the nonlinear function of the data distribution Φ corresponds to conditions $\sum_{k=1}^N \Phi(x_k) = 0$, and \tilde{k}_t is the average value of kernel matrix $\mathbf{K} \in R^N$ ($[\mathbf{K}]_{ij} = [k(\mathbf{x}_i, \mathbf{x}_j)]$). This matrix consists of kernel vectors $\mathbf{k}_t \in R^N$, $[\mathbf{k}_i]_j = [k_t(\mathbf{x}_i, \mathbf{x}_j)]$, and is calculated according to modified rule $\mathbf{k}_t(\mathbf{x}_i, \mathbf{x}_t) = \left\langle \rho_{j,i}^{x_j} (1 - \rho_{j,i}^{x_j}) \right\rangle$, where ρ indicates empirical coefficients, selected from the model of the studied phenomena [8].

If we apply this filter and remember to reconcile the sets of data, we can offer the form of the covariance matrix P^a to analyze the actual errors based on the form of the covariance matrix of the extrapolated value of the state vector of the P^f system and the matrix of covariance of the observational data R :

$$P^a = P^f - P^f A^T (A P^f A^T + R)^{-1} A P^f. \quad (47.25)$$

Thus, we obtain a tool for optimizing the calculation of the matrix of optimal coefficients of the Kalman amplification and, accordingly, correction of the extrapolation values of the vector of the true state of the system with regard to the aggregate data of modeling and observations.

After completing the data integration procedure, we can calculate the necessary parameters by the algorithm [8]:

$$x_t^{(ij)} = \sum_{m=1}^n w_{ij}(\tilde{x}_t^m) x_t^m, \quad (47.26)$$

where $w_{ij}(\tilde{x}_t^m)$ is the weighting coefficient, determined by the rule [8, 13]:

$$\min \left\{ \sum_{m=1}^n \sum_{x_t^m \in R^m} w_{ij}(\tilde{x}_t^m) \left(1 - \frac{x_t^m}{\tilde{x}_t^m} \right)^2 \right\}. \quad (47.27)$$

In this equation, m is the number of experiments conducted; n is the number of data sources; x_t^m is the distribution of the results of observations; R^m is the set of data; \tilde{x}_t^m is the corrected extrapolation values of the vector of the true state of the system by aggregate modeling and observational data.

Therefore, we obtain a regular spatial distribution of measurable characteristics in the local study area, both as a result of model calculations, and taking into account regional measurements on a grid that corresponds to the distribution of data, that is, it has a much better resolution than the usual model.

47.4 Approach to the Risk Assessment as the Analysis of Coherent Complex Measures of Multidimensional Multivariate Distributions

The key question of the assessment is the quantification of risk in cases of multidimensional multivariate variables. This case requires a correct assessment of all parts of the loss function. The set of corresponding probabilities p , that is, the sets of quantiles q corresponding to the distribution p should be analyzed and estimated. The distribution function can be continuous (and in this case the function $q(p)$ under the conditions of continuous distribution of p should be analyzed), or discrete (and in this case we obtain a set of N separate values $q(p)$ for each p). In cases where the distributions of the measured quantities and the distributions of the corresponding probabilities p are not known and cannot be represented as linear normal distributions (in most real cases), nonparametric estimation methods should be used.

Nonparametric methods are aimed at risk assessment without applying rigorous assumptions about the properties of the distributions studied. That is, instead of the input of distribution parameters, we estimate the risk by empirical distributions, which stem from the properties of the distributions of data. In relation to parametric methods, nonparametric methods can prevent the risk of a false a priori estimation of distribution, which leads to large errors in the final risk measurements. Nonparametric methods are based on the assumption that the closest values are distributed in the same way as the previous data; thus, we can predict the evolution of the system using the latest data (represented by empirical distributions). The usefulness of nonparametric methods in practice depends on the observance of this assumption in different situations. As practice shows, in most cases this assumption is fulfilled and, thus, nonparametric methods work well. On the other hand, if this assumption is not fulfilled, especially at the boundaries of distributions, nonparametric methods may have sensitive uncertainties, in particular, if distributions have extremes.

Typically, quantile values are derived from histograms of empirical data distributions, which are regularized and specified in various ways. In particular, it is possible to regularize the histograms using kernel methods, which are the most complex of modern nonparametric methods for estimating distributions, and allow smoothing of the toothed histograms without introducing rigorous assumptions about the properties of data distributions.

It is also possible to extend the range of nonparametric estimation methods by including not only empirical statistics known from observations, but also additional data, in particular, artificially constructed distributions. For example, it is possible to construct a separate hypothetical scenario

based on individual data, with a certain probability of its implementation, and to apply a nonparametric method of joint assessment of the empirical (“historical”) and hypothetical (“nonhistorical”) scenario. Inclusion of the consideration of additional, for example, hypothetical distributions, allows us to overcome the main disadvantage of nonparametric estimation methods – their complete dependence on the sets of empirical data and, thus, the weakness in determining extremes. An important tool for studying the mutual dependencies of sets of distributions is the copula function. In Frees and Valdez [22] a general description of the use of copula functions for quantitative risk assessment is presented. These methods are described in more detail in Cherubini et al. [23].

For the assessment of long-term multidimensional interrelations in multivariate distributions, based on the results of the data analysis, it is suggested that the copula might be used in an elliptic form [23]. In particular, based on the analysis of the obtained data, for the further analysis of risks and the definition of strategies for minimization of disaster losses, use of a multidimensional probability distribution function can be proposed that takes into account the identified multivariate internal dependencies in the structure of disaster distributions in the best possible manner.

If the observation distribution can be described by random vector (X_1, X_2, \dots, X_n) with mutual distribution function $F_{X_1}(x_1) F_{X_2}(x_2), \dots, F_{X_n}(x_n)$ and marginal distribution functions $F_{X_1}(x_1) F_{X_2}(x_2), \dots, F_{X_n}(x_n)$, on the hypercube $[0, 1]^n$ a distribution function C with the uniform marginal distributions on the segment $[0, 1]$ can be defined, and the condition $F_{X_1, X_2, \dots, X_n}(x_1, x_2, \dots, x_n) = C(F_{X_1}(x_1), \dots, F_{X_n}(x_n))$ will be fulfilled.

Application of the regularization algorithms allowed the distributions satisfying the stated conditions to be obtained; in addition, the distribution functions F_{X_i} may be presented as continuous for all $i = 1, 2, \dots, n$, and, thus, the distribution function C , which is called the copula, will be defined unambiguously. For our case, the copula can be represented as:

$$C_{v,c}(u_1, u_2, \dots, u_n) = \frac{v\sqrt{|\mathbf{c}|} (v\pi)^n}{v+n} \cdot \frac{\prod_{n=1}^N \frac{(1+\frac{1}{n})^{\frac{v+n}{2}}}{1+\frac{v+n}{2n}}}{\prod_{n=1}^N \frac{(1+\frac{1}{n})^{\frac{v}{2}}}{1+\frac{v}{2n}}} \cdot \int_{\mathbf{x}_1}^{(u_1)} \dots \int_{\mathbf{x}_n}^{(u_n)} \left(1 + \frac{\mathbf{x}_i^{(x,y)}}{\mathbf{c}\mathbf{v}}\right) d\mathbf{x}_i^{(x,y)}, \tag{47.28}$$

where \mathbf{c} is the correlation matrix obtained from analysis of regularized data.

The possibility of the application of this type of copulas to describe the dependencies between the disaster risks is confirmed by studies in the field of finance and insurance,

in particular, Bradley and Taqqu and Rachev et al. [24, 25]. Thus, the proposed formula can be considered with a particular case of the general approach to risk assessment described in Rachev et al. [25].

In this form, we get the opportunity to determine the long-term trends of the interconnected processes and phenomena represented by stochastic observational data [26, 27].

However, one should take into account the fact that the risks in complex cases of describing large multicomponent, multiphysics systems at certain time intervals cannot be adequately described by linear superpositions of scalar correlations [28]. Here, it is necessary to use more complex dependencies, which reflect the complex interconnections in the system, and take into account the spatial-temporal heterogeneity and uncertainty inherent in the studied phenomena.

For analysis of the studied phenomena in intervals, in which its behavior differs essentially from normal, we propose using the following copula [29]:

$$C(u_1, u_2) = \exp\left(-V\left(-\frac{1}{\log u_1}, -\frac{1}{\log u_2}\right)\right), \tag{47.29}$$

$$V(x, y) = \int_0^1 \max\left(\frac{\omega}{x}, \frac{1-\omega}{y}\right) dH(\omega), \tag{47.30}$$

where:

$$H(\omega) = \begin{cases} 0, & \omega < 0; \\ 1/2(\omega(1-\omega))^{-1-\alpha}(\omega^{-\alpha}(1-\omega)^{-\alpha})^{\frac{1}{\alpha-2}} d\omega, & 0 \leq \omega < 1; \\ 1, & \omega \geq 1 \end{cases} \tag{47.31}$$

For analysis of interdependent (or weakly dependent) phenomena, for example, hydrological disasters, we can use the form $0 \leq \omega < 1$.

This formalization allows a better understanding of the interdependencies between climatic parameters and disaster distribution on a regional scale, and additionally allows regularization algorithms for reducing uncertainty to be integrated [30].

For further analysis of risk behavior measurement dependent on number of climatic, ecological, etc., independent heterogeneous parameters, we propose another algorithm. This method is based on the approach to coupled nonparametric analysis of multidimensional, multivariate distributions by kernel copulas [31]. Using this approach it is possible to reduce uncertainties and errors connected with differences in measurement intervals, and to smooth gaps in data distribution [32].

If $K_{u,h}(x)$ is the kernel vector for $u \in [0;1]$ on interval $h > 0$ we can propose accordingly [31]:

$$K_{u,h}(x) = \frac{K(x) (a_2(u, h) - a_1(u, h) x)}{a_0(u, h) a_2(u, h) - a_1^2(u, h)}, \quad (47.32)$$

$$a_l(u, h) = \int_{\frac{u-1}{h}}^{\frac{u}{h}} t^l K(t) dt, \quad l = 0, 1, 2. \quad (47.33)$$

Also, in this case the functions $G_{u,h}(t)$ and $T_{u,h}$:

$$G_{u,h}(t) = \int_{-\infty}^t K_{u,h}(x) dx, \quad (47.34)$$

$$T_{u,h} = G_{u,h}\left(\frac{u-1}{h}\right) \quad (47.35)$$

can be defined.

The distribution function of the complex parameter will be determined by distribution functions of studied parameters X_1, X_2, \dots, X_n using copula C:

$$F(x_1, x_2, \dots, x_n) = C(F_1(X_1), F_2(X_2), \dots, F_n(X_n)). \quad (47.36)$$

Distribution of extremes of the studied parameters will be described by distribution functions $F_i(x)$ corresponding to threshold $x_i > u_i$ as:

$$\hat{F}_i(x) = 1 - \frac{N_{u_i}}{n} \left(1 + \hat{\xi}_i \frac{x - u_i}{\hat{\beta}_i}\right)^{-\frac{1}{\hat{\xi}_i}}, \quad i = 1, 2, \quad (47.37)$$

where ξ indicates the smoothing parameter, and β indicates interdependence parameter ($\beta \in [0,1]$; $\beta = 0$ for independent distributions, and $\beta = 1$ for absolutely dependent distributions).

In this case, the optimal kernel copula estimator may be presented as [8, 33]:

$$\hat{C}(u, v) = n^{-1} \sum_{i=1}^n G_{u,h}\left(\frac{u - \hat{F}_1(X_{i1})}{h}\right) G_{u,h}\left(\frac{v - \hat{F}_2(X_{i2})}{h}\right) - (uT_{u,h} + vT_{u,h} + T_{u,h}T_{v,h}). \quad (47.38)$$

For example, for the area studied in northwestern part of Ukraine, on the base of multiyear statistics an “optimal correlator” between air temperature and disaster frequency was determined: “reduced max temperature” [8]:

$$T_{red} = \left(1 - \frac{\frac{1}{N} \sum_{n=1}^N T_n}{T_{max}}\right) \left(1 - \frac{1}{T_{max} - \frac{1}{N} \sum_{n=1}^N T_{max}}\right). \quad (47.39)$$

Here, N is the number of meteorological measurements, T_n is the measured air temperature, T_{max} is the maximum registered air temperature. This parameter allows more accurate interrelations between the long-term change of climatic indicators and the frequency of hydrological and climatic disasters to be calculated. The approach proposed is more correct relative to the analysis with traditional variables. Depending on the time interval, the multicomponent correlation obtained allows increased accuracy of the assessment of disaster frequency up to 22% [8]. This is an essential value for mid- and long-term regional forecasting.

47.5 Notes on Decision-Making Approaches in Environmental Risk Analysis Using Satellite Data

Despite a well-developed theory, risk assessment remains a complex task with deep uncertainties in many important cases. Especially in such complex cases as socio-environmental tasks, when we are dealing with complex multicomponent and multiphysics systems heterogeneously distributed in time and space.

In the general sense, the risk of R causing damage or potential losses L as a result of disaster occurring with probability p can be estimated as:

$$R_i = L_i p(L_i). \quad (47.40)$$

Or for all assemblage of events $i \in I$ on the interval I : $R = \sum_i L_i p(L_i)$.

In the integral form [34] it is possible to present:

$$R(x, t) = \int f_i(x, \omega) P(x, d\theta), \quad i = 1, \dots, m, \quad (47.41)$$

where $f(x, \omega)$, and $P(x, \omega)$ indicates stochastic functions and random distributions of probable events, which lead to a negative impact. Thus, in the general case, the risk is described as a superposition of several random processes.

In the general case, the definition of the functions $f(x, \omega)$ and $P(x, \omega)$ is a very complex task, which requires the introduction of additional conditions and restrictions. But

application of satellite observations and modeling can help to determine these functions.

Let us interpret the components of the equation of risk $f(x, \omega)$ as a function of propagation of the natural process, which produces hazardous effects on the components of the natural and anthropogenic environment in the considered area, and $P(x, \omega)$ as a function of the impact of individual components and objects of natural and anthropogenic environment in the considered territory under the influence of dangerous influences, according to Ermoliev et al. and Pflug and Werner [34, 35].

Under such an assumption, the risk assessment in each individual case can be reduced to two general cases. In the first case, we accept the impact function of the distribution of vulnerable components over the observation area (for example, vulnerable infrastructure, ecosystem component, etc., obtained from satellite observation), in a certain way ranked by the disaster statistics, and focus on the analysis, and on the verification of the set of features derived from the problem-oriented model of the drivers of a dangerous process.

Otherwise, we confine ourselves to the general definition of the properties of the propagation function, and concentrating on the analysis of the impact function, and on the statistical properties of the distribution of indicators of influence that we can observe. Depending on the specific problem being solved, the first, second path, or a combined approach can be used.

Regarding the methods of integrated information gathering, it should be noted that during the last 40 years, remote sensing and especially satellite observation methods have demonstrated explosive progress, including significant development of its methodology [36, 37]. As long as the basic methodological idea is now based on models of signal formation, the monitoring methods are based on the models of individual indicators, and control of natural processes is based on the analysis of sets of indicators [36, 38]. Our understanding of ecosystem structure and functioning is now more comprehensive and accurate: the complex nexus between the processes and phenomena can be analyzed, the feedback in an multi-agent environment can be simulated, and the integrated dynamics of the processes can be modeled to predict the behavior of multicomponent systems [39]. In this context, the remote sensing methods play a role as a source of information about the behavior of important variables of the used interlinked models. Therefore, remote sensing is now the tool not only for monitoring but also for predictions and forecasts.

Therefore, further evolution of our ecological knowledge, and the development of remote sensing technologies, require further development of the methodology of remote sensing applications, including new methods of data processing and

interpretation, a new foundation for the selection of the sets of interrelated indicators, based on the new ecological models [40]. Utilization of new kinds of indicators may also require new approaches to risk assessment, based on advanced methodology. In particular, this kind of approach should not be based on assessments of mean values and their deviations [41], but should be focused on optimal decision making in a complex multicomponent and multiphysics environment.

Let us consider a formal task. To solve the task of remote sensing data utilization in a multimodel approach to risk assessment for decision making in a complex multicomponent and multiphysics environment as a formal representation, we first need to consider the problem of the models' application to the selection of the optimal set of remote sensing indicators [8].

The initial stage includes the process of forecast generation, which is based on the set of initial a priori assumptions, observed or measured values [42], described as a vector $x = (x^1, x^2, \dots, x^s)$.

The further step is the modeling: model recalculation of these values into a group of core hydrological, biophysical, and climatological series (with $F(x, \varepsilon, y) = 0$, $(x, \varepsilon) \rightarrow y$), collected in a vector y . Next, based on the information from the pair (x, y) , we calculate values for a list of parameters, grouped into what is referred to as the vector of satellite indicator-based models: $z = (z^1, z^2, \dots, z^s)$ (with $z^s = g^s(x, y)$, $(x, y) \rightarrow z^s$).

As a result of the integrated modeling we obtain parameters summarized in the triplet (x, y, z) . This combined vector is a starting point for the modeling of socio-economic, socio-ecological, and risk parameters, as presented in Fig. 47.2 [42].

Usually, we consider a group of S satellite data-based models, labeled $s \in \{1, 2, \dots, s\}$. Each of these equations is such that the endogenous variables z^s can be obtained as an explicit mapping of the core variables $z^s = g^s(x, y)$.

Therefore, the satellite model might be conceptually presented by the time-series (x, y, z) , which will determine the behavior of z_t^s , with a residual term μ_t^s :

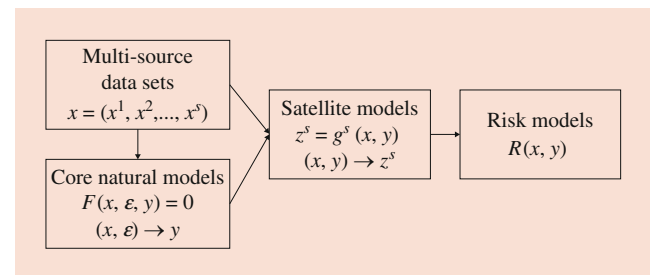


Fig. 47.2 Approach to decision making and risk assessment using multi-source and multimodel data in tasks of socio-environmental security

$$z_t^s = f(x_t, x_{t-1}, \dots, x_{t-L}, y_t, y_{t-1}, \dots, y_{t-L}, z_{t-T}^s, \mu_t^s). \quad (47.42)$$

This presentation of the satellite model in the simple unidirectional time-series form has neither interactions with other variables nor any feedback between z_t^s and the core initial a priori assumptions [40]. Among the wide range of satellite data-based models, the time-series models are well known and widely used, for example, in the form of autoregressive moving average models including, for example, autoregressive lags and/or moving-average components. To find the most usable model of the data-generating process for a given risk metric Z_t a number of standard methodologies may be applied:

$$Z_t = c + \sum_{l=0}^N \beta_l X_{t-l} + \sum_{l=0}^P \rho_l Y_{t-l} + \sum_{l=1}^L \partial_l Z_{t-l} + \sum_{k=0}^K \theta_k \varepsilon_{t-k}. \quad (47.43)$$

In this equation Z_t is a satellite variable vector, X_t is a vector of initial core variables, Y_t is a row vector of the layer of core parameters, ε_t is the stochastic error, and c , β , ρ , and ∂ are the unknown parameters.

It should be noted that in some cases, the approach is sensitive to the used method: for example, using autoregressive methods in the model may lead to a decrease in the impact of the core drivers on the target variables [43]. Therefore, depending on risk metric Z_t and on the type of core and supplementary variable the varied form of Eq. (47.43) may be applied. In particular, for the analysis of climate-related risk an approach based on copula utilization may be used [8].

In the modern paradigm of earth sciences complex risk analysis requires involvement of statistics from a combination of ecology, climatology, hydrogeology, hydrology, and geo-statistics to produce a correct analysis of the statistical properties of the estimated model [8, 42]. Therefore, a variable selection for identification of core drivers of the dynamic behavior of the studied socio-ecological risk variable is a key aspect of satellite model development [8, 42].

It should also be noted that the sensitivity of the approach to the method is varied on different time and spatial scales. For example, despite impressive results of recent years and good fitting of the existing data, models based on data mining, machine learning, or neural networks often fail in long-term forecasts, because they are not able to follow a changing external environment because they lack a theoretical basis. At the same time, because machine learning in environmental security can be interpreted as a nonparametric method of assessment of the high dimensional systems, the proposed approach may also be utilized as a way of adapting the machine-learning methods to separate tasks. It makes sense to assume that the best analytical and prediction approaches

should be based on a combination of statistics and physical models, and should combine geo-ecological models with statistical optimization [42]. Besides, models built this way have the additional benefit of ease of interpretation.

To develop a satellite model the optimal exogenous drivers X_t , Y_t should be selected from a set of all potential drivers in Eq. (47.43). Potential drivers can usually be defined based on relevant theory or a priori assumptions and identified by calibration measurements. Application of calibration measurements ensures that the most robust model for the tested variables is obtained. The selected drivers should be statistically significant and have the known distribution parameters; to obtain a required distribution a regularization procedure should be applied [8, 42]. The conditional dynamic forecasts of Z_t can be generated based on the selection and estimation of the satellite model, taking into account the sets of final parameter estimates and the forecasts of the core variables from the first stage. The validation of the final model on the number of samples is the final estimation step. The final models selected by the search procedure are reviewed for consistency with initial assumptions.

Thus, the problem is the selection of variables for each model type $(x, \varepsilon) \rightarrow y$, the search for the relevant type of formal relationship $(x, y) \rightarrow z^s$ between the physical and observable variable parameter, and the development of the total distribution for each type of risk investigated [42]. Directed modeling should solve this problem. The problem of regularization of the initial distribution of variables should be considered separately [42].

Therefore, using the proposed method the sets of parameters can be obtained, describing the state of the system analyzed. Further, based on the obtained parameter distributions, it is possible to estimate a distribution of risk and make management decisions.

Let us consider a task on the methodology of risk assessment, based on the optimal decisions, and based on the obtained sets of indicators [41]. First of all, it should be noted that in the formal task of risk assessment and management, the reduction or non-increase of losses should be used as a quantitative parameter of control. At the same time, the variables affecting the characteristics of the management or decision-making system can be both controlled and unmanageable. In this situation the controlled variables should be described as the parameters of decision making in the context of the influence of input data on the behavior of unmanageable variables. The full cycle of collecting, processing, interpretation of information about the system studied, the decision making, and the analysis of the system's response to decisions may be considered within the framework of the "information-response" formalization. The "information-response" formalization can be presented as follows [9]. Let us define as presumably stochastic $I(x, y, z)$ input information obtained from measurements, observations,

and model forecasts; $H_I(i|\theta)$ is the probability distribution function, where θ is a state of the studied object or system. Because the state of the system cannot be determined with certainty, the appropriate probability distribution $p(\theta)$ and distribution $H_I(i|\theta)$, which describe a priori incompleteness of information available on the studied system, should be defined.

Let us define as the decision function $d(I)$ a formalized response to incoming information, presented as the making and implementation of management decisions. Let us also define losses as $l(d(I), \theta)$ under conditions of a constant state of the studied systems θ , or with a defined change in this state. The expected losses associated with the expansion of dangerous processes, connected to the management decisions based on the information received, described by the decision function d , can be presented as [44]:

$$\begin{aligned} R(I, d) &= R\left[H_I(\bullet), d\right] \\ &= \iint l(d(i), \theta) dH_I(i|\theta) p(\theta) d\theta. \end{aligned} \quad (47.44)$$

To minimize this risk the optimal Bayes decision function d^* can be applied:

$$R(I(z^s), d^*) = \min_{d(i)} \iint l(d(i), \theta(x, y)) dH_I(i|\theta) p(\theta) d\theta. \quad (47.45)$$

Thus, it is formally shown that the risk can be reduced to a simple decision function. At the same time it is important that the minimization of losses requires a completeness of information on the studied system. In other words, all states θ should be determined, for each of which a solution $\{a\}$ can be defined, which form a set of possible decisions A .

Let us further consider the realization that the set of data i^* , which consists of information obtained from direct measurements, observations, and model forecasts, optimizes the decision function d^* , and can be applied to minimize risk. This set of data nominally makes information (I) formally completed (I^*). Formally, we can assume that the information is nominally full if there is a separate realization of information I or a single state of the studied system that meets all of the set of data; or, formally speaking: $I^* \equiv I$ if there exists a function $\phi(i)$, that when $H_I(i|\theta) \neq 0$, $\theta = \phi(i)$. In the case analyzed it means that the set of models $(x, \varepsilon) \rightarrow y$ operated the sets of parameters $(x, y) \rightarrow z^s$, controlled by certain technological tools within the framework of a sustainable methodology of measurement, should be developed. Taking into account the above-described models, these requirements correspond to the satellite models z_i^s .

Basing on the requirement for information completeness, an optimal decision function can be as follows:

$$d^*(i^*) = b, l(b, \phi(i^*)) = \min_a l(a, \phi(i^*)). \quad (47.46)$$

In this formal case, with an appropriately defined optimal decision function and nominally complete information, the risk function can be defined as:

$$R(I^*, d^*) = \int \min_{a \in A} l(a, \theta) p(\theta) d\theta. \quad (47.47)$$

Although it is important to remember that this approach is sensitive to the mutual completeness of information: for the analysis of behavior of the distribution of $H_I(i|\theta)$ and $p(\theta)$ and so to the determination of the realization of i^* of set I , the relevant models should be developed.

This formalization allows a wide range of optimal decision functions to be utilized: a stochastic [45], Bayesian [44], or fuzzy operator [46], depending on the task, data availability, and properties of their distributions can be applied.

The proposed approach is a complex of models and algorithms, which is aimed at the calculation of a unique set of parameters that should be obtained from observation systems, using determined tools for data processing and interpretation. Equation (47.47), based on an important nexus, allows the distributions of risk to be estimated and a basis for decision making to be developed too.

Speaking of modern methods of statistical analysis of multisource data in the tasks of environmental risk analysis, it should be noted that the modeling of geo-systems and the remote sensing interpretation methods should be an integral part of the risk assessment systems. We need to increase our level of knowledge in the field of earth sciences, as well to increase requirements in the area of decision making, because new challenges define new methodological frameworks. The approach proposed is based on the methodology that allows the problem definition of using the satellite observations in tasks of socio-ecological security to be expanded. In particular, in addition to traditional statistical analysis directed at change detection, it is possible to predict a state of the studied systems, basing on the models of geo-systems. These capabilities certainly expand the scope and sphere of the application or approach, and could positively affect the reliability of the results obtained through the use of different sources of data. Also important is that the proposed methodology includes feedback between management decisions and the system state, and it is formally postulated that risks depend on the decision made and on past, current, and planned management impacts on the system. Therefore, such a complex of methods and algorithms, combined into common methodology, could positive affect the

effectiveness of management decisions and the quality of risk assessment [41].

47.6 Conclusions and Discussion

A nonparametric, multistage algorithmic procedure for the statistical analysis of observational, measurement, and statistics data, which allows regularized distributions to be obtained in units invariant to the nature of the data, is proposed and discussed in this chapter. It allows input datasets for risk modeling to be obtained, and at the same time the different types of emergencies to be analyzed, irrespective of their origin, despite the different time and spatial scales that are inherent to driving forces and processes that generate danger.

In addition, the fundamentals of the regionalization of climate change models based on regularized regional observations are proposed through the integration of local meteorological observations into global climate models based on the Kalman filtration method. This approach allows multi-scale modeling to be utilized, with multisource data in risk assessment tasks.

Thus, it can be argued that a methodological basis for assessing the long-term risks of disasters associated with climate and environmental change has been proposed. That is, a methodology for the estimation of long-term socio-ecological risks based on integrated monitoring is proposed, and an approach to mathematical modeling of the environment using satellite data is given. This approach can be used as a methodological basis for decision making in the field of socio-ecological security.

At the same time, some important limitations should be mentioned.

The proposed method of multistage analysis of disaster parameters, based on the KPCA algorithm has certain limitations arising from the nature of the algorithm used.

First, the method relates to the so-called local modeling methods [19], and thus to the family of nonparametric estimation methods based on the properties of the studied data distributions, and do not require the inclusion of additional conditions, a priori distribution parameters, etc. In general, local models, unlike “global” balance models, do not require the implementation of conservation laws [19], or at least, their simultaneous execution.

This allows the freedom of choosing the boundary conditions, not imposing additional complex constraints, and using the entire set of observational data.

At the same time there is another problem. The conservation laws are based on the assumption of the homogeneity of time, and by abandoning the laws of conservation, we lose the time scale, which can be significant for scenario calculations. This problem can be solved by including an artificial timeline for each (or for certain sets) distribution and modeling case,

or, as in our algorithm, by entering time as a universal measure of random sets of observational data. In this way, we are removing the general problem. Regularization of varied sets of heterogeneous data can be correctly performed using the proposed path.

However, when analyzing the distribution of extreme indices in relation to the changes in the mean value, the problem of time arises again in a different form. Using time as a universal measure does not allow distributions of mean and extreme indicators to be analyzed independently in time. That is, within the framework of the presented form of the proposed algorithm it is impossible to accurately determine the short-term gradients of extreme values under the monotonic behavior of the mean.

In practice, this means that effective analysis of the extreme parameters as the driving forces of disasters using the proposed algorithm is possible, usually on a long and on a medium scale. For short-term intervals, increasing the reliability is possible only by improving the spatial resolution of the method.

For the short-term (from several days to several weeks) estimates, the method should be modified, in particular, through verification by local measurement (from a methodological point of view – through the involvement of a quasi-independent set of data with a definite time scale).

The mentioned limitations should be taken into account in application of the proposed method.

Thus, although within the framework of the solved problem (construction of methodology for medium- and long-term risk assessments) the proposed method is quite suitable and is, according to individual indicators, better than existing analogs, it has limitations in its application in solving other types of tasks and requires some modifications.

Acknowledgments The authors are grateful to anonymous referees for constructive suggestions that resulted in important improvements to the chapter, to colleagues from the International Institute for Applied Systems Analysis (IIASA), from the American Statistical Association (ASA), American Meteorological Society (AMS), and from the International Association for Promoting Geoethics (IAPG) for their critical and constructive comments and suggestions. The authors express special thanks to the National Academy of Sciences of Ukraine for partial support of this study within the framework of the multilateral program “Integrated robust management of food-energy-water-land use nexus for sustainable development.”

References

1. Freeman III, A.M., Herriges, J.A., Kling, C.L.: *The Measurement of Environmental and Resource Values: Theory and Methods*. Routledge (2014)
2. Oves, M., Khan, M.Z., Ismail, I.M.: *Modern Age Environmental Problems and their Remediation*. Springer International Publishing (2018)
3. Dalby, S.: *Environmental Security*, vol. 20. University of Minnesota Press (2002)

4. Fortlage, C.A.: *Environmental Assessment: A Practical Guide*. Routledge (2017)
5. Barnett, J.: *The Meaning of Environmental Security: Ecological Politics and Policy in the New Security Era*. Zed Books (2001)
6. Piegorsch, W.W., Xiong, H., Bhattacharya, R.N., Lin, L.: Non-parametric estimation of benchmark doses in environmental risk assessment. *Environmetrics*. **23**(8), 717–728 (2012)
7. Bartell, S.M., Gardner, R.H., O'Neill, R.V.: *Ecological Risk Estimation*. Lewis Publishers, Boca Raton (1992)
8. Kostyuchenko, Y.V.: Geostatistics and remote sensing for extremes forecasting and disaster risk multiscale analysis. In: *Numerical Methods for Reliability and Safety Assessment*, pp. 439–458. Springer International Publishing (2015). https://doi.org/10.1007/978-3-319-07167-1_16
9. Raiffa, H., Schlaifer, R.: *Applied Statistical Decision Theory*. MIT Press/Massachusetts Institute of Technology, Cambridge, MA/London, 356 p (1968)
10. Fowler, H.J., Kilsby, C.G., O'Connell, P.E.: Modeling the impacts of climatic change and variability on the reliability, resilience and vulnerability of a water resource system. *Water Resour. Res.* **39**, 1222 (2003)
11. Cowpertwait, P.S.P.: A generalized spatial–temporal model of rainfall based on a clustered point process. *Proc. R. Soc. Lond. Ser. A*. **450**, 163–175 (1995)
12. Mudelsee, M., Börngen, M., Tetzlaff, G.: On the estimation of trends in the frequency of extreme weather and climate events. In: Raabe, A., Arnold, K. (eds.) *Wissenschaftliche Mitteilungen, Institut für Meteorologie der Universität Leipzig*, vol. 22, pp. 78–88. Institut für Troposphärenforschung e. V. Leipzig, Leipzig (2001)
13. Lee, J.M., Yoo, C., Choi, S.W., Vanrolleghem, P.A., Lee, I.B.: Nonlinear process monitoring using kernel principal component analysis. *Chem. Eng. Sci.* **59**(1), 223–234 (2004)
14. Villez, K., Ruiz, M., Sin, G., Colomer, J., Rosen, C., Vanrolleghem, P.A.: Combining multiway principal component analysis (MPCA) and clustering for efficient data mining of historical data sets of SBR processes. *Water Sci. Technol.* **57**(10), 1659–1666 (2008)
15. Scheolkopf, B., Smola, A.J., Muller, K.: Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.* **10**(5), 1299–1399 (1998)
16. Mika, S., Scheolkopf, B., Smola, A.J., Müller, K.-R., Scholz, M., Ratsch, G.: Kernel PCA and de-noising in feature spaces. *Adv. Neural Inf. Process. Syst.* **11**, 536–542 (1999)
17. Romdhani, S., Gong, S., Psarrou, A.: A multi-view nonlinear active shape model using kernel PCA. In: *Proceedings of BMVC, Nottingham, UK*, pp. 483–492 (1999)
18. Boardman, J.M., Vogt, R.M.: *Homotopy Invariant Algebraic Structures on Topological Spaces*, vol. 347. Springer (2006)
19. Lloyd, C.D.: *Local Models for Spatial Analysis*. CRC Press (2010)
20. Aoki, M.: *Optimization of Stochastic Systems: Topics in Discrete-Time Systems*, vol. 32. Academic (1967)
21. Wang, X., Bishop, C.H.: A comparison of breeding and ensemble transform Kalman filter ensemble forecast schemes. *J. Atmos. Sci.* **60**(9), 1140–1158 (2003)
22. Frees, E.W., Valdez, E.A.: Understanding relationships using copulas. *N. Am. Actuar. J.* **2**(1), 1–25 (1998)
23. Cherubini, U., Luciano, E., Vecchiato, W.: *Copula Methods in Finance*. Wiley (2004)
24. Bradley, B.O., Taqqu, M.S.: Financial risk and heavy tails. In: *Handbook of Heavy Tailed Distributions in Finance*, pp. 35–103. North-Holland (2003)
25. Rachev, S.T., Menn, C., Fabozzi, F.J.: *Fat-Tailed and Skewed Asset Return Distributions: Implications for Risk Management, Portfolio Selection, and Option Pricing*, vol. 139. Wiley (2005)
26. Dupuis, D.J., Tawn, J.A.: Effects of mis-specification in bivariate extreme value problems. *Extremes*. **4**(4), 315–330 (2001)
27. Venter, G.G.: Tails of copulas. In: *Proceedings of the Casualty Actuarial Society*, Vol. 89, No. 171, pp. 68–113. (2002, March)
28. Marti, K., Ermoliev, Y., Makowski, M., Pflug, G.: *Coping with Uncertainty: Modeling and Policy Issues*, vol. 581. Springer Science & Business Media (2006)
29. Genest, C., Ghoudi, K., Rivest, L.-P.: Discussion of “Understanding relationships using copulas” by Edward Frees and Emiliano Valdez. *N. Am. Actuar. J.* **2**(3), 143–149 (1998)
30. Juri, A., Wuthrich, M.V.: Copula convergence theorems for tail events. *Insur. Math. Econ.* **30**, 405–420 (2002)
31. Chen, S.X., Huang, T.: *Nonparametric Estimation of Copula Functions For Dependence Modeling*. Technical Report. Department of Statistics, Iowa State University, Ames, 20 p (2010)
32. Embrechts, P., Lindskog, F., McNeil, A.: Modeling dependence with copulas and applications to risk management. In: Rachev, S. (ed.) *Handbook of Heavy Tailed Distributions in Finance*, pp. 329–384. Elsevier (2003)
33. Kalnay, E., Kanamitsu, M., Kistler, R., Collins, W., Deaven, D., Gandin, L., Iredell, M., Saha, S., White, G., Woollen, J., Zhu, Y., Leetmaa, A., Reynolds, R., Chelliah, M., Ebisuzaki, W., Higgins, W., Janowiak, J., Mo, K.C., Ropelewski, C., Wang, J., Roy, J., Dennis, J.: The NCEP/NCAR 40-year reanalysis project. *Bull. Am. Meteor. Soc.* **77**, 437–470 (1996)
34. Ermoliev, Y., Makowski, M., Marti, K.: *Managing Safety of Heterogeneous Systems: Decisions under Uncertainties and Risks*, vol. 658. Springer Science & Business Media (2012)
35. Pflug, G., Werner, R.: *Modeling, Measuring and Managing Risk*. World Scientific (2007)
36. Elachi, C., Van Zyl, J.J.: *Introduction to the Physics and Techniques of Remote Sensing*, vol. 28. Wiley (2006)
37. Lillesand, T., Kiefer, R.W., Chipman, J.: *Remote Sensing and Image Interpretation*. Wiley (2014)
38. Qiu, H.L., Sanchez-Azofeifa, A., Gamon, J.A.: *Ecological applications of remote sensing at multiple scales*. In: *Functional Plant Ecology*, 2nd edn. CRC Press (2007)
39. Suter II, G.W.: *Ecological Risk Assessment*. CRC Press (2016)
40. Campbell, J.B., Wynne, R.H.: *Introduction to Remote Sensing*. Guilford Press (2011)
41. Ermoliev, Y., von Winterfeldt, D.: Systemic risk and security management. In: *Managing Safety of Heterogeneous Systems*, pp. 19–49. Springer, Berlin/Heidelberg (2012)
42. Kostyuchenko, Y.V.: On the methodology of satellite data utilization in multi-modeling approach for socio-ecological risks assessment tasks: a problem formulation. *IJMMS*. **3**(1), 1–8 (2018)
43. Engle, R.F., Russell, J.R.: Autoregressive conditional duration: a new model for irregularly spaced transaction data. *Econometrica*. **66**, 1127–1162 (1998)
44. Kostyuchenko, Y.V., Yuschenko, M., Kopachevsky, I., Artemenko, I.: Bayes decision making systems for quantitative assessment of hydrological climate-related risk using satellite data. In: Das, K., Ram, M. (eds.) *Mathematical Modelling of System Resilience, “Mathematical and Engineering Sciences” Series*, pp. 113–141. River Publishers, Aalborg (2019)
45. Kopachevsky, I., Kostyuchenko, Y.V., Stoyka, O.: Land use drivers of population dynamics in tasks of security management and risk assessment. *Int. J. Math. Eng. Manag. Sci.* **1**, 18–24 (2016)
46. Kostyuchenko, Y.V., Sztoyka, Y., Kopachevsky, I., Artemenko, I., Yuschenko, M.: Multisensor satellite data for water quality analysis and water pollution risk assessment: decision making under deep uncertainty with fuzzy algorithm in framework of multimodel approach. In: *Remote Sensing for Agriculture, Ecosystems, and Hydrology XIX*, vol. 10421, p. 1042105. International Society for Optics and Photonics (2017)



Yuriy V. Kostyuchenko, Ph.D., D.Sc., Associated Professor, received an MS in Astrophysics in 1993, a Ph.D. in Geophysics and Remote Sensing in 1993, a Ph.D. in Geophysics and Remote Sensing in 1997, and a D.Sc. in Engineering and Remote Sensing in 2018. His research interests include remote sensing and GIS application for environmental and homeland security, risk analysis using nonparametric approaches, and climate and disaster study for food and water security.



Probabilistic Models for Reliability Analysis Using Safe-Life and Damage Tolerance Methods

48

Xuefei Guan and Jingjing He

Contents

48.1	Introduction	965
48.2	Probabilistic Modeling	966
48.2.1	Probabilistic Parameter Estimation – Inverse Problem.....	967
48.2.2	Probabilistic Prediction: Forward Problem.....	968
48.3	Reliability Model	968
48.3.1	Safe-Life Model.....	969
48.3.2	Damage Tolerance Model.....	971
48.4	Application	974
48.4.1	Reliability and Life Prediction of a Curvic Coupling...	974
48.5	Summary	977
	References	978

of interest can be obtained using probabilistic prediction. Next, the reliability model based on the probabilistic modeling is introduced, where the safe-life model and the damage tolerance model are discussed in detail. The life prediction given a certain risk constraint and the time-dependent probability of failure estimation can be made using the developed method. Two examples are employed to demonstrate the overall method.

Keywords

Reliability · Probabilistic models · Safe-life · Damage tolerance · Probability of failure · Bayesian

Abstract

The chapter presents a systematical method for probabilistic analysis using safe-life and damage tolerance models. In particular, the reliability analysis incorporating those models are developed to provide a basic framework for the life prediction given risk constraints and the time-dependent probability of failure estimation. First, the probabilistic modeling is presented, and the uncertainties from model prediction and data are considered. The uncertainties are quantified and are encoded in the probability density functions of model parameters using probabilistic parameter estimation. The propagation of the characterized uncertainties to the result of quantity

48.1 Introduction

The design philosophies have undergone a paradigm shift over the past few decades. The investigation of the disintegration of de Havilland Comet commercial jet in flight in 1950s made it clear that the life estimate of fatigue critical structures are highly necessary. This requirement leads to the safe-life design method. The design practice started to change in the 1960s toward the fail-safe design, which essentially is a principle to provide redundant load paths as backups in the event of localized failure. The fail-safe design can be viewed as an extended version of the safe-life design. In safe-life approach, the fatigue life of a structure or a component is certified by testing data and analysis. A fundamental assumption in this approach is that the material has no initial flaw at the beginning of the service. Although the existence of material flaws depends on the scale at which the material is examined, it is usually determined at a scale that a standard inspection technique can be applied. For example, a turbine disk is considered flawless when no indication is found using the certified ultrasound inspection. An indication is defined as the echo amplitude along the sound path being larger

X. Guan (✉)
Graduate School of China Academy of Engineering Physics, Beijing, China
e-mail: xfguan@gscap.ac.cn

J. He
School of Reliability and Systems Engineering, Beihang University, Beijing, China
e-mail: 09722@buaa.edu.cn

than a predefined threshold, e.g., 20% of the screen height. This predefined threshold is used to differentiate a small flaw from the backscattered noise components in the echo signal data caused by material grain boundaries. As a result, material flaws producing smaller echo amplitudes cannot be effectively identified from the inspection data. Indeed in certain circumstances, these flaws can be picked up using special techniques such as micro-CT in the lab environment; but time, cost, and environmental constraints restrict those special techniques being used in a more general setting. Damage tolerance design method, on the other hand, considers the material of a component or a structure has initial natural flaws at the beginning of the service. The failure of F-4 Phantom II due to the fracture of the fail-safe wing structure in 1973 signified that a structure could not be fully fail-safe without inspection. This failure event together with the following DC-10 commercial jet landing crash at Sioux city in 1989 promoted the development of damage tolerance analysis, and the development finally spawned the certification of damage tolerance for life-limiting components in aircraft structures and turbine engines [1, 2].

Reliability analysis entails the computation of the failure probability under various sources of uncertainties. For mechanical components whose major failure modes are dominated by fatigue and fracture, uncertainties can arise from materials, manufacturing, service conditions, and so on. Due to the inherent nature of those uncertainties, it is difficult to completely eliminate all uncertain sources. The propagation of uncertainties from input or sources to the output leads to uncertain results; therefore, uncertainty modeling and quantification play an important role in risk mitigation for critical engineering infrastructures. Although the nature of uncertainties has been discussed in many fields for a long time, from a pragmatic point of view, whether an uncertainty is aleatory (intrinsic) or epistemic (lack of knowledge) can be less interest to an engineering analysis as the uncertainty quantification is made through a mathematical modeling process [3]. The uncertainty due to the lack of knowledge is encoded in the discrete model space where using one or more alternative models becomes subjective. The uncertainty due to intrinsic randomness is encoded in the parameter space of a chosen model or a joint model parameter space when a set of competing models are available. Instead of enforcing a clear label for an uncertainty, it can be more beneficial to treat the overall uncertainty as the model uncertainty and parameter uncertainties associated with models in a hierarchical manner for the ease of quantification [4]. This chapter is focused on the safe-life and damage tolerance models in the context of uncertainties; in particular, the uncertainties are dealt with in a probabilistic inference framework.

48.2 Probabilistic Modeling

Following the convention random variables are denoted using capital letters, and their samples are denoted using lowercase letters. Denote Y_i as a proposition asserting that the i th data value is in the range of y_i and $y_i + dy_i$. Similarly denote Z_i and E_i as propositions asserting that the model prediction for the i th data value is in the range of z_i and $z_i + dz_i$ and that the i th error value is in the range of e_i and $e_i + de_i$, respectively. The actual data value y_i and model prediction z_i can be expressed as

$$y_i = z_i + e_i. \quad (48.1)$$

Denote the prediction value of a mathematical model M as $m(x_i|\theta)$ where x_i is the model independent variable and $\theta \in \Theta$ is the uncertain model parameter vector. Using $p(\cdot)$ to represent the probability distribution, the probability distributions for propositions Z_i and E_i are expressed, respectively, as

$$p(Z_i|M, \theta) = f_Z(z_i), \quad (48.2)$$

and

$$p(E_i|M, \theta) = f_E(e_i). \quad (48.3)$$

Given Z_i and E_i are independent the probability distribution for proposition Y_i can be expressed as

$$p(Y_i|M, \theta) = \int \int p(Z_i|M, \theta)p(E_i|M, \theta)p(Y_i|Z_i, E_i, M, \theta) dZ_i dE_i. \quad (48.4)$$

From Eq. (48.1)

$$p(Y_i|Z_i, E_i, M, \theta) = \delta(y_i - z_i - e_i). \quad (48.5)$$

Substitute Eq. (48.5) into Eq. (48.4) to obtain

$$p(Y_i|M, \theta) = \int f_Z(z_i)f_E(y_i - z_i)dz_i. \quad (48.6)$$

For a *deterministic model prediction*, $z_i = m(x_i|\theta)$ and $f_Z(z_i) = \delta(z_i - m(x_i|\theta))$. In this case Eq. (48.6) writes

$$\begin{aligned} p(Y_i|M, \theta) &= \int \delta(z_i - m(x_i|\theta))f_E(y_i - z_i)dz_i \\ &= f_E(y_i - m(x_i|\theta)). \end{aligned} \quad (48.7)$$

When the error term is a Gaussian variable, the above equation can be written as

$$p(Y_i|M, \theta) = \frac{1}{\sqrt{2\pi}\sigma_e} \exp \left\{ -\frac{[y_i - m(x_i|\theta)]^2}{2\sigma_e^2} \right\}, \quad (48.8)$$

where σ_e is the standard deviation of the error variable e_i .

For a *probabilistic model prediction* including a statistical error term ϵ_i such as $z_i = m(x_i|\theta) + \epsilon_i$, marginalizing

$$p(Z_i|M, \theta) = \int p(Z_i|\mathcal{E}_i, M, \theta)p(\mathcal{E}_i|M, \theta)d\mathcal{E}_i \quad (48.9)$$

over ϵ_i with $p(Z_i|\mathcal{E}_i, M, \theta) = \delta(z_i - m(x_i|\theta) - \epsilon_i)$ to obtain

$$\begin{aligned} p(Z_i|M, \theta) &= \int \delta(z_i - m(x_i|\theta) - \epsilon_i)f_{\mathcal{E}}(\epsilon_i)d\epsilon_i \\ &= f_{\mathcal{E}}(z_i - m(x_i|\theta)) = f_Z(z_i). \end{aligned} \quad (48.10)$$

Substitute $f_Z(z_i)$ in Eq. (48.6) with Eq. (48.10) and obtain

$$p(Y_i|M, \theta) = \int f_{\mathcal{E}}(z_i - m(x_i|\theta))f_E(y_i - z_i)dz_i, \quad (48.11)$$

which is the convolution of the two probability distributions. It is known that when the error terms are two independent Gaussian variables, the convolution is another Gaussian probability distribution that can be expressed as

$$p(Y_i|M, \theta) = \frac{1}{\sqrt{2\pi}\sqrt{\sigma_e^2 + \sigma_{\epsilon}^2}} \exp \left\{ -\frac{[y_i - m(x_i|\theta)]^2}{2(\sigma_e^2 + \sigma_{\epsilon}^2)} \right\}, \quad (48.12)$$

where σ_{ϵ} is the standard deviation of the statistical error variable ϵ_i .

In addition if the model independent variable x_i is also uncertain with a distribution function $p(X) = f_X(x_i)$ and given that the model $m(x_i|\theta)$ is monotonic and differentiable over the domain of x , it is seen that

$$f_Z(z_i) = f_X(x_i) \left| \frac{dx_i}{dz_i} \right| = f_X(m^{-1}(z_i|\theta)) \left| \frac{dm^{-1}(z_i|\theta)}{dz_i} \right|. \quad (48.13)$$

where $m^{-1}(\cdot)$ is the inverse function. Assume that X follows a normal distribution with a mean of \bar{x}_i and a standard deviation of σ_x , the uncertainty from variable x_i can then be integrated into Eq. (48.6) and obtain

$$\begin{aligned} p(Y_i|M, \theta) &= \frac{1}{\sqrt{2\pi}\sqrt{\sigma_e^2 + \sigma_{\epsilon}^2 + (J\sigma_x)^2}} \\ &\times \exp \left\{ -\frac{[y_i - m(\bar{x}_i|\theta)]^2}{2[\sigma_e^2 + \sigma_{\epsilon}^2 + (J\sigma_x)^2]} \right\}, \end{aligned} \quad (48.14)$$

where $J = \left| \frac{dm^{-1}(z_i|\theta)}{dz_i} \right|$ and \bar{x}_i is the mean value of x_i . For multidimensional independent variable x_i , the term $(J\sigma_x)^2$ in Eq. (48.14) should be replaced with $J\Sigma_x J^T$ where J is the Jacobian matrix and Σ_x is the covariance matrix of x_i .

It is worth mentioning that for a forward analysis, e.g., the analysis of uncertainty propagation from various sources to result y through the model $m(x|\theta)$, the separated qualifications using σ_e , σ_{ϵ} , σ_x or Σ_x , and so on can be explicitly included and used. For inverse problems, e.g., the estimation of the model parameter θ , the detailed uncertainty contributions from different sources are not known a priori; therefore, it can be convenient to consider using a total standard deviation $\sigma_t^2 \equiv \sigma_e^2 + \sigma_{\epsilon}^2 + (J\sigma_x)^2$ and rewrite Eq. (48.14) as

$$p(Y_i|M, \theta) = \frac{1}{\sqrt{2\pi}\sigma_t} \exp \left\{ -\frac{[y_i - m(\bar{x}_i|\theta)]^2}{2\sigma_t^2} \right\}. \quad (48.15)$$

48.2.1 Probabilistic Parameter Estimation – Inverse Problem

The probabilistic parameter estimation can be recast into a probabilistic inference problem. The Bayesian method is a rational approach to perform probabilistic inference using three pieces of information, namely, the prior, the likelihood function, and the data. The basis of the Bayesian method is nothing but the conditional probability and the product rule of probability. The posterior probability distribution function of the model parameter is expressed using the product of the prior and the likelihood as

$$p(M, \theta|Y_i) \propto p(M, \theta)p(Y_i|M, \theta). \quad (48.16)$$

The “proportional to” symbol \propto in the above equation indicates the actual posterior distribution is known up to an normalizing constant

$$p(Y_i) = \sum_{M \in \mathcal{M}} \int_{\theta \in \Theta_M} p(M, \theta)p(Y_i|M, \theta)d\theta, \quad (48.17)$$

where \mathcal{M} is a finite set of available models and Θ_M is the parameter space for θ under model M . The normalizing constant is of no particular interest when the posterior is evaluated using the Markov chain Monte Carlo (MCMC) sampling method with only one model M available. It is needed when multiple models M are compared in the context of Bayesian model assessment. As only one model is considered in this study, the symbol M is omitted hereafter for simplicity. In addition the variable σ_t is also an uncertain variable and is independent of the model parameter θ ; the joint posterior probability density function (PDF) writes

$$p(\theta, \sigma_i | Y) \propto p(\theta, \sigma_i) p(Y | \theta, \sigma_i). \quad (48.18)$$

Given no prior information on (θ, σ_i) , the Jeffreys' noninformative prior [5], i.e., $p(\theta, \sigma_i) \propto 1/\sigma_i$, can be used. In this case the posterior PDF for a total of n data points $Y = (Y_1, Y_2, \dots, Y_n)$ is expressed as

$$p(\theta, \sigma_i | Y) \propto \frac{1}{\sigma_i} \cdot \frac{1}{(\sqrt{2\pi}\sigma_i)^n} \exp \left\{ -\frac{\sum_{i=1}^n [y_i - m(\bar{x}_i | \theta)]^2}{2\sigma_i^2} \right\}. \quad (48.19)$$

The approximation of the above PDF can be made using samples obtained from MCMC simulations with the standard Metropolis-Hastings (MH) sampling strategy [6, 7]. An alternative approach is using asymptotic approximation. The widely used one is the Laplace method [8, 9]. The basic idea of the Laplace method is to use a multivariate normal distribution to approximate the posterior PDF of Eq. (48.19). Under the condition that the PDF has one mode and is approximately symmetric around the mode, the Laplace method can yield accurate results.

48.2.2 Probabilistic Prediction: Forward Problem

The basis for probabilistic prediction is centered on three pieces of information, namely, a mathematical model correlating the input (independent) variable and the output (dependent) variable, a probabilistic representation of the model parameters, and the input variable. Once the model is chosen and the parameter is characterized in a probabilistic manner, the next step is to propagate the uncertainty through the model to output and quantify the uncertainty in the output results.

A universal method for the purpose is to use Monte Carlo-based method. The idea is to draw many samples of the uncertain variables from its distributions. Each of the samples can be seen as a realization of the randomness, and the corresponding model output associated with that particular sample is a realization of one possible output, following a certain probability distribution. Given that the number of the samples is sufficiently large (in statistical sense), the resulting empirical distribution of the model output ensures a convergence to the true distribution. The mean and confidence intervals of the model prediction can trivially be obtained using the empirical distribution.

A subtle but crucial point that is often overlooked is the differentiation between the mean estimation confidence interval and the prediction confidence interval. Recall the uncertain parameters are (θ, σ_i) and the probabilistic model prediction is

$$y_j = m(\bar{x} | \theta_j) + e_j, \quad (48.20)$$

where θ_j is the j th sample, and e_j is a random sample drawn from the error distribution with a standard deviation of $\sigma_{i,j}$. When using MC estimators for evaluation, the mean confidence interval does not involve the generation of an error variable according to $\text{Norm}(0, \sigma_{i,j})$ in j th realization; however, the prediction confidence interval of an MC estimator requires the error variable. The difference is signified in examples in the next sections.

48.3 Reliability Model

The early and formal definition of the engineering "Reliability" can be found in NASA's report [10]: *the probability that a system, subsystem, component or part will perform its intended functions under defined conditions at a designated time for a specified operating period*. It is a probabilistic measure because of the existence of uncertainty. In practice, the reliability can also be characterized using mean time between failure (MTBF), failure rate, and the probability of failure (POF), depending on the fields of applications and the criteria for decision-making. The probability of failure is widely used in structural reliability analysis and is adopted here. To understand the probability of failure, it is necessary to define a failure criterion for the problem in consideration. The failure criterion is a subjective choice fully tied to the specific engineering applications. For example, a failure can be defined as the fracture of a component due to a crack propagation in one case, but it can also be defined as a fatigue crack being larger than a length far below the one causing a sudden fracture. In other cases, it can also be defined as when a flaw is identified in a part in service using nondestructive inspection. Once the failure criterion is chosen, the response of a model becomes a binary output of failure or success.

Given a failure criterion such that the event of $g(\phi) \leq g_0$ is considered as a failure where $g(\phi)$ is the model taking a parameter ϕ . The so-called response function is then defined as $h(\phi) = g(\phi) - g_0$. The failure criterion essentially divides the entire multidimensional parameter space into two domains, the failure domain and the success domain. The POF is the failure domain portion of the entire domain. Mathematically it is the integral over the failure domain of the performance function and is expressed as

$$P_f \equiv \Pr [h(\phi) \leq 0] = \int_{h(\phi) \leq 0} p(\phi) d\phi. \quad (48.21)$$

Using MC-based methods for POF evaluation, the MC estimator of POF is

$$\hat{P}_f \approx \frac{1}{N_{mc}} \sum_{i=1}^{N_{mc}} \mathbf{1}(\phi_i), \quad (48.22)$$

where $\mathbf{1}(\cdot)$ is the identity function taking the value according to

$$\mathbf{1}(\phi) = \begin{cases} 1 & h(\phi) \leq 0 \\ 0 & \text{otherwise} \end{cases}, \quad (48.23)$$

N_{mc} is the total number of MC samples, and ϕ_i is the i th MC sample of the parameter vector.

48.3.1 Safe-Life Model

The safe-life model for fatigue analysis is usually referred to the Basquin law [11] for stress-life (s - N) data or Manson-Coffin law [12, 13] for strain-life (ϵ - N) data and their variants, considering loading, environmental, work-hardening effects, and so on. The most simple yet useful one is the log-linear model, which presents the strain/stress and the fatigue life as a linear relationship in the log-transformed coordinates. For example, the log-transformed Manson-Coffin low-cycle fatigue life model reads

$$\log N = b_0 + b_1 \log (\Delta \epsilon / 2), \quad (48.24)$$

where N is the number of cycles to failure or fatigue life, $\Delta \epsilon$ is the strain range during one load cycle, and (b_0, b_1) are model parameters to be determined by testing data. For linearized models (log-transformed linear, piece-wise linear, and the combinations of the two), the conventional linear regression can directly be applied. The median curve of the fit, the confidence interval for the median curve, as well as the prediction interval given the input can all analytically be evaluated.

It is worth mentioning the difference between the confidence interval of the median curve and the prediction interval of the model is vital for reliability analysis. For prediction purposes, the latter one should be pursued. Figure 48.1 shows the life testing data, a log-linear model's mean fit, confidence interval of the mean, and the prediction confidence interval. Note that the independent variable of the model is plastic strain amplitude ($\Delta \epsilon_p / 2$) and the dependent variable is the number of cycles to failure (N). The dependent variable is plotted in x -axis only to follow the convention. It can be observed in Fig. 48.1 that the prediction confidence interval is much wider than the confidence interval of the mean. Figure 48.2 shows PDFs of the mean life and predicted life.

Cumulative Damage Models

The log-linear model or the more general linearized model can be used to calculate life (N) under a given strain range ($\Delta \epsilon$); however, a proper treatment is needed when different strain ranges are repeated as a block loading. The fatigue process can be virtually thought as a process of life consumption by each of the strain ranges ($\Delta \epsilon_i, i = 1, \dots, k$). The concept is shown in Fig. 48.3. For one cycle in the block with a strain

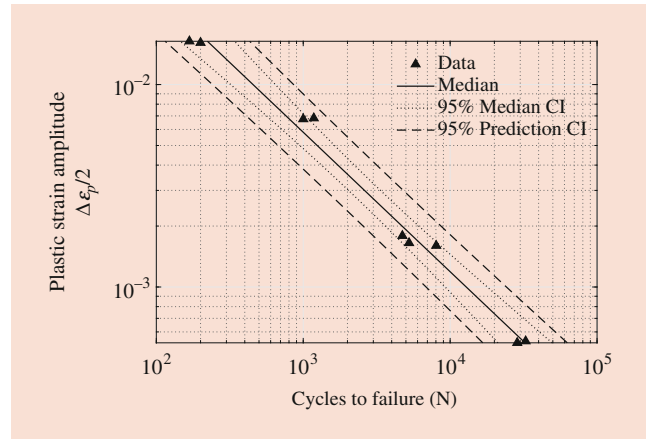


Fig. 48.1 Low-cycle fatigue testing data and the log-linear strain-life model. The life testing data (triangles) [14], the mean (solid line), the 95% confidence interval of the mean fit (dotted lines), and the 95% prediction confidence interval (dashed lines) are shown

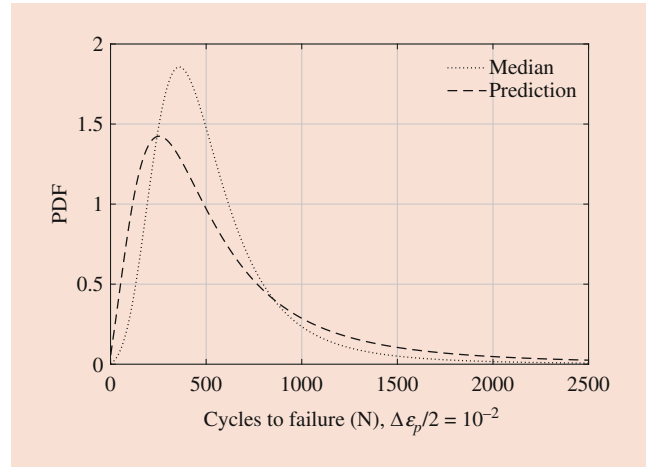


Fig. 48.2 PDFs of the median and the predicted life for $\Delta \epsilon_p / 2 = 10^{-2}$. The life is transformed back to the linear coordinates

range of $\Delta \epsilon_i$, the total useful life is consumed by a fraction of $1/N_i$. Denote the number of cycles associated with the strain range of $\Delta \epsilon_i$ in the block as n_i . The consumed fraction of the total life per one block is $(n_1/N_1 + n_2/N_2 + \dots + n_k/N_k)$ and the total number of cycles in one block is $(n_1 + n_2 + \dots + n_k)$. The equivalent consumed fraction of the total life per one cycle is

$$\Delta d = \frac{\frac{n_1}{N_1} + \frac{n_2}{N_2} + \dots + \frac{n_k}{N_k}}{n_1 + n_2 + \dots + n_k} = \sum_i^k \frac{f_i}{N_i}, \quad (48.25)$$

where $f_i = n_i / \sum_{i=1}^k n_i$ is the percentage (or frequency) of $\Delta \epsilon_i$ in the series of load cycles and N_i is the fatigue life at strain range $\Delta \epsilon_i$. Consequently the equivalent life in terms of cycles writes

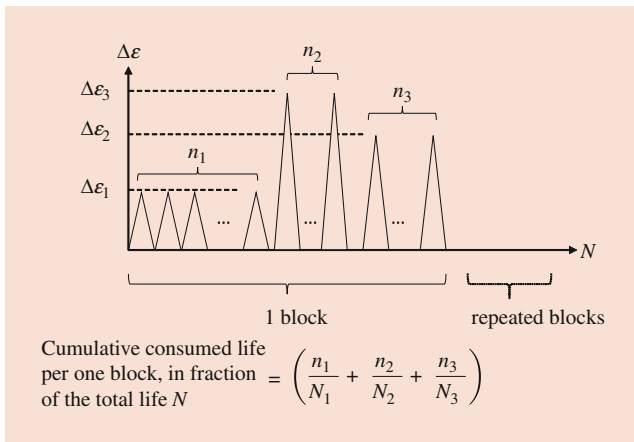


Fig. 48.3 Illustration of the repeated load block consisting of load cycles with different strain ranges

$$N_{eq} = \frac{1}{\Delta d} = \left[\sum_{i=1}^k \frac{f_i}{N_i} \right]^{-1}. \tag{48.26}$$

Equation (48.25) is the linear damage rule (Palmgren–Miner rule) originally suggested by Palmgren [15] and popularized by Miner [16]. It is later observed that the load sequence alters the accumulation and many variants accounting for the load sequence effects have been proposed. A comprehensive survey of cumulative damage models can be found in Ref. [17].

The aforementioned models describe the deterministic relationship between stress/strain range with the fatigue life, as well as the treatment for block loads. A probabilistic model can be established given a deterministic model and PDFs of the model parameters identified using experimental data. To demonstrate the probabilistic safe-life model with the linear damage rule, a repeated load block consisted of $\Delta\epsilon_1 = 5 \times 10^{-3}$ and $\Delta\epsilon_2 = 10^{-3}$ is considered. The frequency ratio is $n_1/n_2 = 1/3$. The log-linear model in Eq. (48.24) and a Gaussian likelihood with an unknown standard deviation σ_t are used. The Bayesian posterior of (b_0, b_1, σ_t) with a noninformative prior (i.e., $\propto 1/\sigma_t$) writes

$$p(b_0, b_1, \sigma_t) \propto \frac{1}{\sigma_t} \cdot \frac{1}{(\sqrt{2\pi}\sigma_t)^n} \times \exp \left\{ -\frac{\sum_{i=1}^n [\log \bar{N}_i - b_0 - b_1 \log(\Delta\bar{\epsilon}_i/2)]^2}{2\sigma_t^2} \right\}, \tag{48.27}$$

where $(\bar{\epsilon}_i, \bar{N}_i)$, $i = 1, \dots, n$ are experimental data. Here the data in Fig. 48.1 (solid triangles) are used. MCMC simulations are employed to draw samples from the Bayesian

posterior. The Metropolis-Hastings random walk [6, 7] is adopted with an initial values of (b_0, b_1, σ_t) set to be the linear regression mean $(-0.1, -1.42, 0.23)$. A total number of 10^7 samples are obtained and the results are shown in Fig. 48.4.

The prediction of the life under the given block load can be made with the MCMC samples. For each of the samples, a random realization can be obtained using Eq. (48.26) and Eq. (48.24). With all the MCMC samples, a total of 10^7 random realizations of the equivalent fatigue life are obtained. Based on the 10^7 realizations of the equivalent life, the empirical CDF of N_{eq} is obtained and presented in Fig. 48.5. Based on the empirical CDF of N_{eq} , the equivalent life under a probability of failure (POF) can be evaluated using the MC quantile estimator. For example, the fatigue life associated with a POF of 10^{-3} is the 0.001 quantile of the samples for N_{eq} . In addition, the asymptotic confidence interval for the MC quantile estimator can also be obtained using central limit theorem (CLT) as [19, 20]

$$N_{p,1-\alpha} = \hat{N}_p \pm z_{\alpha/2} \cdot \frac{\sqrt{p(1-p)}}{f(\hat{N}_p)\sqrt{N_{mc}}}, \tag{48.28}$$

where p is the probability (e.g., $p = 0.001$), \hat{N}_p is the MC estimation of p -quantile, N_{mc} is the number of samples, $f(\hat{N}_p)$ is the PDF of the life evaluated at \hat{N}_p , and $z_{\alpha/2} = \Phi^{-1}(1 - \alpha/2)$ and $\Phi^{-1}(\cdot)$ is the inverse CDF of the standard normal variable. For example, if a 95% = $1 - \alpha$ confidence interval is chosen, $\alpha = 0.05$ and $z_{\alpha/2} = 1.96$. It should be noted that the PDF $f(\cdot)$ is not known; therefore, a proper method to estimate $f(\hat{N}_p)$ should be used. Experiences have shown that the kernel density estimator (KDE) method reported in Ref. [18] can produce reliable and fast estimation using the samples.

It should be stressed that the confidence interval is associated with the MC estimator, reflecting the sampling uncertainty of the Monte Carlo method. For example, a 95% confidence interval loosely means that there is 95% chance that an individual MC estimation result of the POF falls in the interval. On one hand the plot in Fig. 48.6 allows for estimation of POF at a given time in terms of equivalent cycles; on the other, given a prescribed POF, the usable time in terms of equivalent cycles can be retrieved. For instance, if the requirement for reliability is $POF = 10^{-5}$, the 95% confidence interval of the allowable life in terms of equivalent cycles is 266~310. The fatigue life in safe-life design practice for many life-limiting parts such as turbine disks and shafts refers to the crack initiation life. As in these applications the goal is to avoid crack propagation stage for safety purposes; therefore, fatigue testing must be carefully designed to support the expectation. The criterion for initiation may vary depending on the applications. For turbine disks, and shafts, the crack initiation refers to an

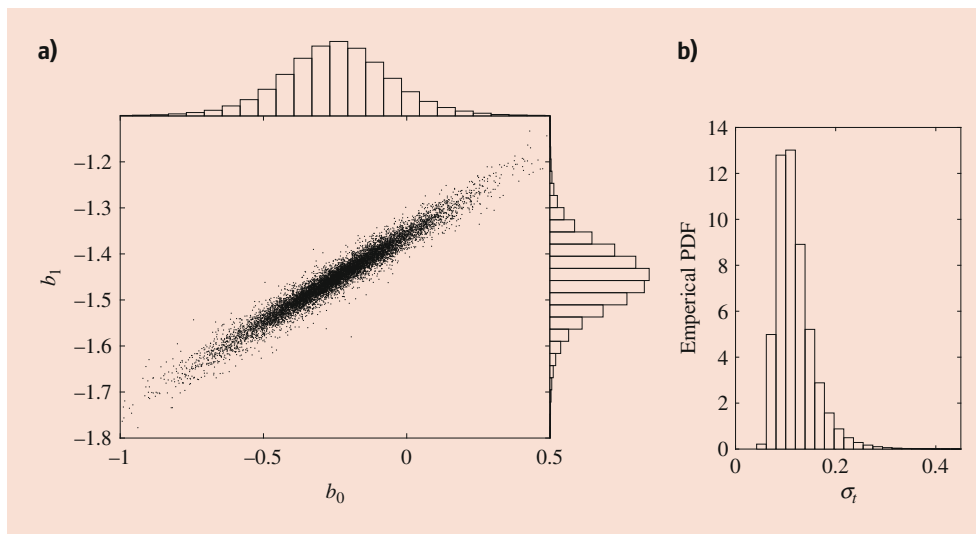


Fig. 48.4 MCMC samples drawn from Eq. (48.27). (a) Samples and histograms of model parameters b_0 and b_1 , and (b) histogram of the standard deviation σ_r in the Gaussian likelihood

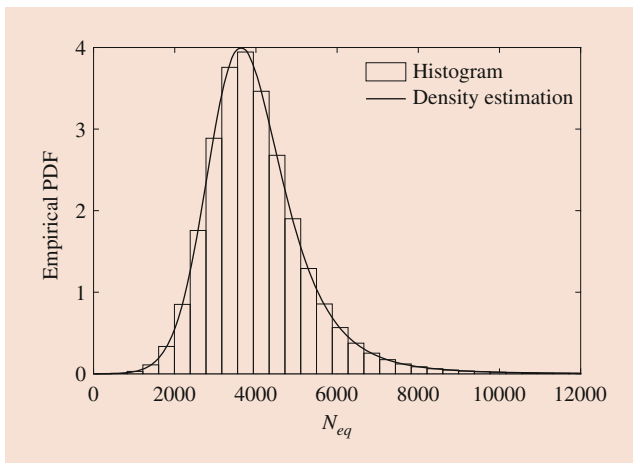


Fig. 48.5 Prediction results of the equivalent life under the given block load. The density estimation result in solid line is made based on the fast kernel density estimation method in [18]

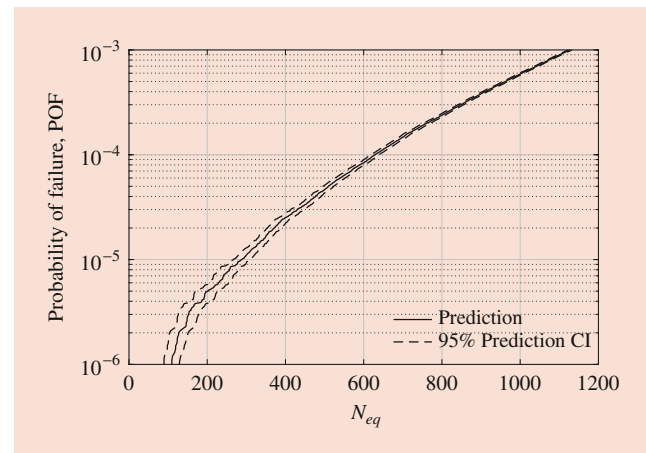


Fig. 48.6 Prediction results of the probability of failure (POF) under the given block load. Results are obtained using 10^7 MCMC samples. The median and confidence interval of the Monte Carlo (MC) estimator are shown

engineering crack size (0.38 mm radius) [21], which is tightly associated with the capacity of the nondestructive inspection for the parts. Conducting the fatigue testing of a specimen until it breaks is not unusual since such a termination of the testing is convenient; but this testing data includes both crack initiation life and crack propagation life and should be used with special care. Experiences have shown that for particular materials the crack propagation life is roughly 1/3 of the fracture life; therefore, the 2/3 of the fracture life can be used as a rough estimate for the crack initiation life. It is noted that this type of special treatment is not universal for all materials, but such a procedure does find its applications in safety-critical components [22, 23].

48.3.2 Damage Tolerance Model

Damage tolerance is the ability of the structure to resist failure due to the presence of flaws, cracks, or other damage, for a specified period of unrepaired usage. The size of the flaw is usually determined using the 90/95 probability of detection (POD) curve offered by the nondestructive testing (NDT) technique used in component manufacturing and in-service inspection. The size is also called 90/95 flaw size, indicating there is 95% confidence that at least 90% of all flaws of this size would be detected. For example, the recommended 90/95 flaw sizes of aircraft engine components are 0.035-inch diameter and 0.035 by 0.070 inches for embedded and

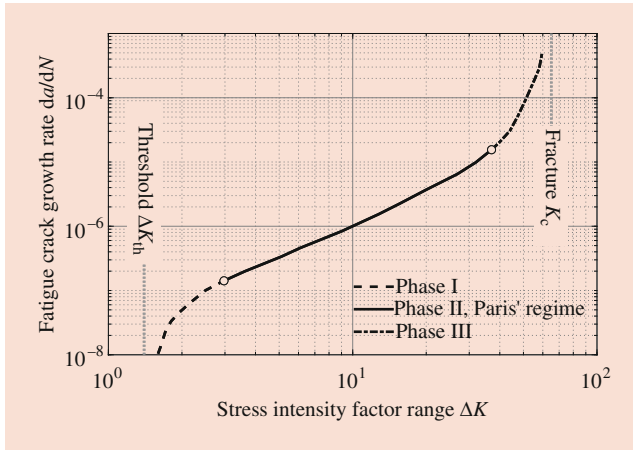


Fig. 48.7 The process of fatigue crack growth. The stable growth phase, also called Paris' regime, can be described by the log-linear Paris' equation. Here the actual values for ΔK and da/dN are just for illustration purposes

surface flaws, respectively [24]. For conservativeness a flaw is treated as a sharp fatigue crack with the worst orientation and shape with respect to the local stress field. The crack can propagate under cyclic loads to reach a critical size causing a sudden fracture. The total duration measured in the number of load cycles or equivalent operation hours is the fatigue crack growth life. The main focus of damage tolerance analysis is to evaluate the remaining useful life given an existing flaw. The fatigue crack growth is one of the major damage mechanisms for mechanical components; however, other damage mechanisms such as creep crack growth, corrosion, and wear are also commonly seen in realistic applications. The cyclic variation of stress intensity factor (SIF) is the main driving factor for fatigue crack growth. As shown in Fig. 48.7, the fatigue crack growth process can be divided into three phases. The widely used Paris' model establishes the log-linear relationship between the crack growth rate and the SIF range in the stable growth phase (Paris' regime) [25]:

$$\frac{da}{dN} = C (\Delta K)^m, \quad (48.29)$$

where a is the crack size, N is the number of load cycles, ΔK is the SIF variation during one load cycle, and C and m are model parameters. Take logarithm on both sides to obtain the log-linear model, allowing for linear regression on crack growth testing data.

For constant amplitude cyclic load, Paris' equation performs well in the stable phase of the crack growth. Realistic service load spectra contains variable amplitude load cycles, and the effect of a load spike can alter the crack growth rate significantly; therefore, the concept of ΔK_{eff} is proposed to include the effect of crack growth retardation [26–28]. The load ratio for constant amplitude load cycles also has

an impact on the fatigue crack growth rate over the entire stable phase of the crack growth and can be accounted for by introducing the load ratio parameter R into the Paris' equation [29–31]. For spectrum loading without load interactions or overload/underload effects, the cumulative damage rule in previous section can be applied:

$$\frac{da}{dN_{\text{eq}}} = \sum_{i=1}^k \omega_k \cdot C (\Delta K_k)^m, \quad (48.30)$$

where $\omega_k = n_i / \sum_{i=1}^k n_i$ is the frequency of k th cyclic stress range in the repeated load block, as illustrated in Fig. 48.3. To consider the load interaction and load ratio effects, appropriate variants of Paris' equation should be used.

In this chapter the Paris' equation is adopted for methodology demonstration without loss of generality. Figure 48.8 presents Virkler's [32] fatigue growth testing data on aluminum alloy in (a) and log-linear fitting on da/dN vs. ΔK in (b). To obtain the da/dN and ΔK , the following finite difference scheme is used [33]. $da_j = a_j - a_{j-1}$, $dN_j = N_j - N_{j-1}$, and $\Delta K_j = \sqrt{\pi \bar{a}_j} \Delta \sigma \cdot g(a, W)$. The index term $j = 2, \dots, q$ where q is the total number of a vs. N data points of a crack growth trajectory, W is the width of the specimen, $\Delta \sigma$ is the stress amplitude of the remote loading, $g(a, W)$ is the geometry correction factor, and $\bar{a}_j = (a_j + a_{j-1})/2$ is the average crack size of two adjacent measurement points. For a center through-thickness crack $g(a, W) = \sqrt{\sec(\pi \bar{a}_j/W)}$. More detailed geometry and loading information of the testing data can be found in Ref. [32].

Using Virkler's crack growth data, the parameters of Paris' equation are obtained using linear regression with da/dN vs. ΔK as a whole in log-transformed coordinates. The mean and covariance matrix of $(\ln C, m)$ are $\mu_p = [-26.155, 2.874]$, $\Sigma_p = \begin{bmatrix} 0.00235, & -0.00038 \\ -0.00038, & 0.000063 \end{bmatrix}$, respectively. The standard deviation of the error variable is $\sigma_t = 0.236$. It is known that in linear regression on $\ln(da/dN)$ vs. $\ln \Delta K$, the error variable is an addition term to the log-linear model. However, the error variable is transformed into an multiplication coefficient to the crack growth rate when the log-linear equation is transformed back to the regular coordinates for cycle-based integration of the crack growth trajectory. Consequently the analytical formulation of linear model prediction is only valid in the log-transformed space.

To perform the prediction for crack growth trajectory, the MC estimator is employed. A total number of 10^7 samples are drawn from the multivariate normal distribution with mean μ_p and covariance matrix Σ_p . For each of the samples the crack growth trajectory is evaluated, and the median prediction of the crack growth trajectory and median prediction confidence interval are estimated from the resulting 10^7 trajectories. For prediction confidence interval considering

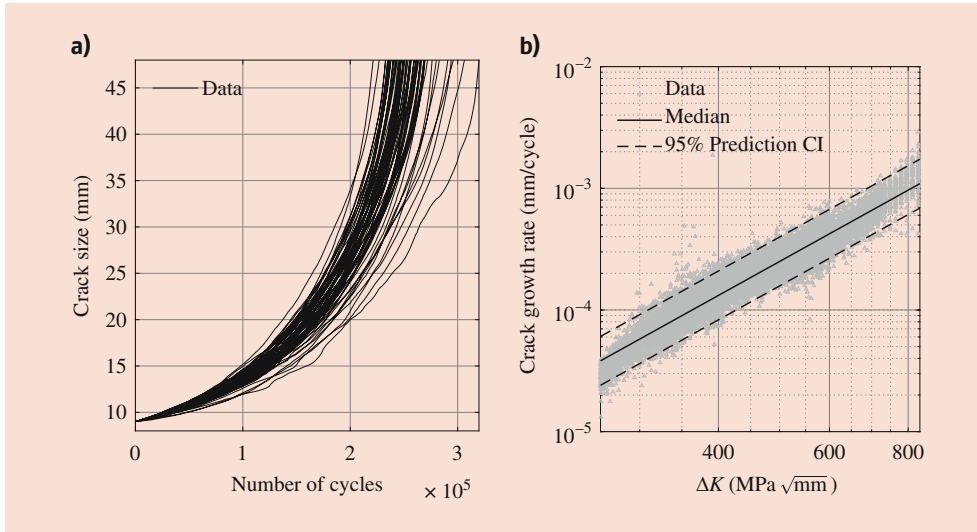


Fig. 48.8 Virkler’s fatigue testing data. (a) Crack size vs. number of cycles and (b) da/dN vs. ΔK and the log-linear fitting result. da/dN is obtained using discrete a vs. N data points associated with each of the crack growth trajectories in (a)

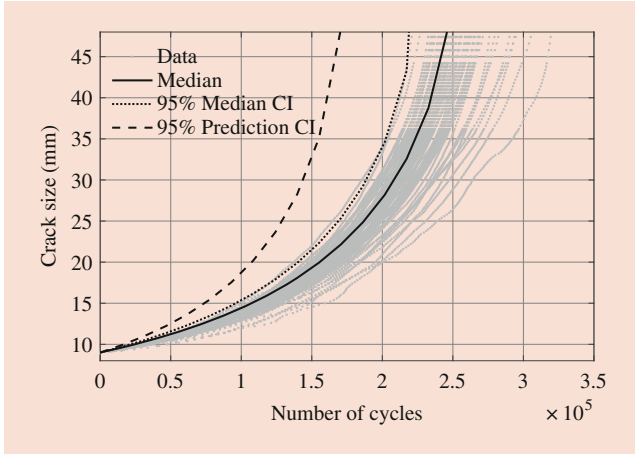


Fig. 48.9 Median and predicted crack growth trajectories for Virkler’s dataset

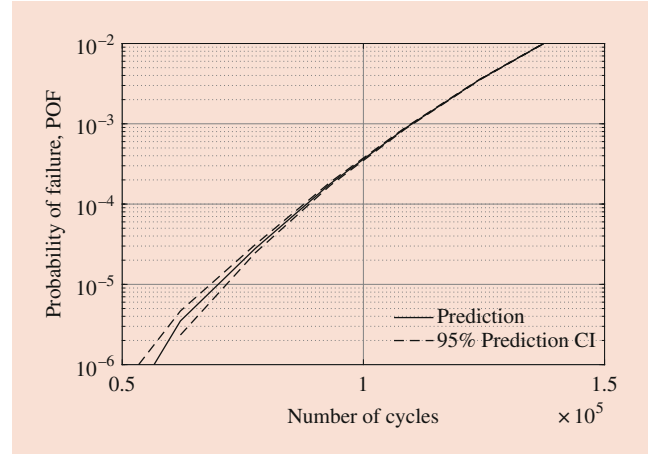


Fig. 48.10 Prediction results of the probability of failure (POF) using Virkler’s dataset. The failure is defined as the crack size being larger than 40 mm. Results are obtained using 10^7 MC samples. The median and confidence interval of the Monte Carlo estimator are shown

the modeling error, a number of 10^7 samples for the error variables are drawn from the Gaussian distribution with a zero mean and a standard deviation of σ_r . For each of the samples $(\ln C, m, \epsilon)$, the a multiplication coefficient $\exp(\epsilon)$ is applied to the right-hand side of Eq. (48.29), and the crack growth trajectory is evaluated. The prediction confidence interval is estimated using the resulting 10^7 prediction trajectories. Figure 48.9 presents the results obtained using the above MC estimator.

Given the definition of the failure event, e.g., the crack size being larger than 40 mm, the POF estimation is made using the MC estimator Eq. (48.22). Figure 48.10 shows the time-dependent POF results for Virkler’s dataset. The confidence interval of the MC estimator for rare event probabilities reads

$$\widehat{P}_{f,1-\alpha} = \widehat{P}_f \pm z_{\alpha/2} \sqrt{\text{Var}(\widehat{P}_f)}, \quad (48.31)$$

where $z_{\alpha/2}$ is defined as before in Eq. (48.28), \widehat{P}_f is given in Eq. (48.22), and $\text{Var}(\widehat{P}_f)$ is the variance of the MC estimator given by

$$\text{Var}(\widehat{P}_f) = \frac{1}{N_{mc} - 1} \left[\frac{1}{N_{mc}} \sum_{i=1}^{N_{mc}} \mathbf{1}(\phi_i)^2 - \widehat{P}_f^2 \right]. \quad (48.32)$$

The term $\mathbf{1}(\phi_i)$ is the identity function defined in Eq. (48.23), N_{mc} is total number of MC samples, and ϕ_i is the i th MC sample $(\ln C, m, \epsilon)$. It is worth mentioning that for rare event

probability estimations, a sufficient number of samples are required to yield a small relative error (RE) in the results. This is due to the fact that only when the number of sampling is large enough the tail region (small probability) of the distribution can be sufficiently sampled. A quick estimate for the required number of samples for rare event simulations can be made using the following equation [34]:

$$\text{RE}_{\text{mc}} \stackrel{\text{def}}{=} \frac{\text{Var}(\hat{P}_f)}{P_f} \approx \sqrt{\frac{1}{N_{\text{mc}} \cdot P_f}}. \quad (48.33)$$

For example, the MC estimator for a rare event with a probability of 10^{-5} needs a number of 10^7 to achieve a relative error about 10%.

48.4 Application

Shaft couplings are key components connecting different parts and transmitting torques in rotating machines such as power transmission systems and turbine engines. Among all the shaft couplings, Curvic couplings are widely used in aeroengines, heavy-duty gas turbines and coaxial drive systems due to its precise centering ability and excellent load carrying capacity. The failure of Curvic couplings can directly affect the overall structural integrity of a rotating machine. The startup and shutdown of the rotating machines form the mechanical load for Curvic couplings, and the repeated startup and shutdown form the cyclic fatigue load. As life-limiting parts in turbine machines, the assessment of the fatigue life and reliability of Curvic couplings must be carefully made to ensure the operation safety. The prediction of the fatigue life of Curvic couplings requires the material properties and loading information. Due to the complex geometry and the contact working condition of Curvic couplings, direct measurements of the loading information such as stress and strain in a realistic working condition is inaccessible. Finite element method (FEM) and/or boundary element method (BEM) have been used to evaluate the stress and strain profiles of Curvic couplings. Existing studies have shown that the state of contact surface can affect the overall stress and strain distribution around the contact surfaces of the contacting parts. To evaluate fatigue life and reliability of Curvic couplings, a full understanding of the stress and strain state of the coupling incorporating the manufacturing uncertainty is necessary. In addition, uncertainties from surface roughness and material must be scientifically quantified.

A full three-dimensional contact FE model of a Curvic coupling assembly in an aeroengine is employed for stress and strain analysis. Typical rotation and loading conditions are applied as boundary conditions, representing the realistic operation environments. The surface uncertainty introduced in the manufacturing process is modeled using the coefficient

of friction of contacting surfaces of the Curvic coupling, e.g., concave tooth/convex tooth. Transient analysis is performed to obtain the time-dependent stress and strain profile formed in a simplified duty cycle of the engine. A linear model is used to correlate the coefficient of friction and the maximum strain range in the duty cycle, and the variation of surface state can propagate to the variation of the maximum strain range at critical locations of the Curvic coupling. A probabilistic low-cycle fatigue model considering the surface effect is established by incorporating the surface parameter into the classical Manson-Coffin model.

48.4.1 Reliability and Life Prediction of a Curvic Coupling

The overall modeling process is presented in Fig. 48.11. The first component is to obtain the surface-dependent stress/strain results. The second component is the probabilistic life prediction module which considers the material uncertainty, surface uncertainty, numerical modeling uncertainty, and so on. The manufacturing-induced surface uncertainty is represented using the variation of the coefficient of friction (ρ) as shown in the figure. Using different values of the parameter ρ , the numerical analysis results of the Curvic coupling are obtained using transient contact FE models. Results are subsequently used to build the correlation model between the coefficient of friction (ρ) and the maximum cyclic strain/stress range ($\Delta\varepsilon$). This correlation is then incorporated into the probabilistic life prediction module for reliability and fatigue life estimation.

Stress and Strain States of Curvic Couplings Subject to Surface Uncertainty

The transient analysis of the assembly of a Curvic coupling is a complex modeling process. To perform an effective numerical analysis, the boundary conditions and the load cycle must be carefully set to represent the realistic working condition. Manufacturing uncertainties can effect on the overall efficiency, performance, and life of the engine, and it has become an important issue in the aircraft engine manufacturing industry [35]. Constrained by metal cutting conditions, the uncertainty in the process will affect the surface roughness of the components and further affect the friction coefficient of contact surface [36,37]. In particular, the surface uncertainty introduced in the manufacturing process is modeled using the coefficient of friction (ρ). The coefficient of friction can affect the contact model of the tooth-tooth contacting of the Curvic coupling. Consequently, the deformation of the contact teeth and the resulting strain and stress distributions over the teeth are altered. The transient analysis is performed for the Curvic coupling assembly with the following condition shown in Table 48.1. The configuration of the transient

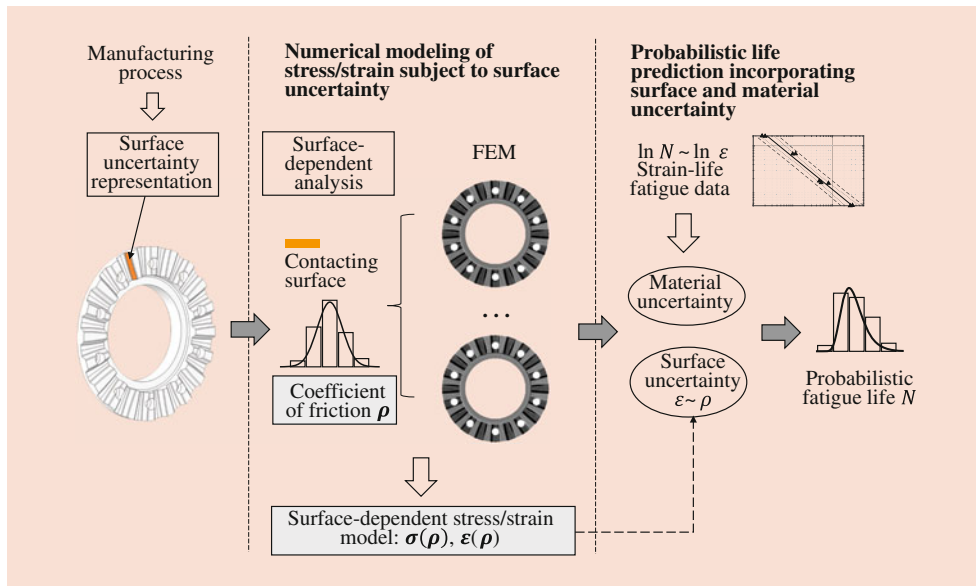


Fig. 48.11 The overall modeling process

analysis represents a simple startup-shutdown cycle of the rotor. In the time duration of 1–10 s, a linear ramp up of the rotational speed from 0 to 2094 rad/s is applied to realize the speed up process of the rotor. A constant moment is applied to the Curvic coupling to simulate the load. The ferritic steel alloy material CrMoV with a Young's modulus of 200 GPa, a Poisson's ratio of 0.3, and a density of $7.85 \times 10^3 \text{ kg/m}^3$ is used. To model the influence of the surface uncertainty on the stress and strain of the Curvic coupling, a set of discrete values for the coefficient of friction (ρ_i) are uniformly chosen in the range of [0, 0.3] with a step size of 0.05. For each of the values in (ρ_i), a transient contact FE analysis is performed using the finite element package ANSYS mechanical workbench. Ten-node quadratic tetrahedron elements are used in the mesh generation as shown in Fig. 48.12a. The average element length is 1.7 mm with program-controlled curvature adaption. A total of seven numerical cases are performed, and the results of the transient stresses and strains are obtained. The location where the strain range is maximized during 0–10s is considered as the fatigue critical region as shown in Fig. 48.12c. To consider the variation in all the contact pairs of the Curvic coupling, the maximum ranges for stress and strain in the repeated sectors shown in Fig. 48.12b are all obtained. The maximum ranges for stress and strain are obtained in one numerical analysis. The results of the maximum strain ranges are shown in Fig. 48.13 as scatter dots. For each ρ value shown in x-axis, a total of ten points from the repeated sectors are plotted. From the results shown in Fig. 48.13, the maximum strain range $\Delta\varepsilon$ and the coefficient of friction ρ show a linearity. Therefore, the following linear model can be built to correlate the coefficient of friction and the maximum strain range:

Table 48.1 The transient analysis setting for the Curvic coupling

Time	Rotational velocity (rad/s)	Moment ($N \times m$)
0	0	540
1	0	540
10	2094	540

$$\Delta\varepsilon = \beta_0 + \beta_1 \cdot \rho + \epsilon_\beta, \quad (48.34)$$

where β_0 and β_1 are fitting parameters and ϵ_β is the error term describing the difference between the model and the actual data. Using linear regression, the mean and covariance matrix of $\beta = [\beta_0, \beta_1]$ are obtained as

$$\mu_\beta = [1.9215 \times 10^{-3}, -4.7177 \times 10^{-4}], \quad (48.35)$$

and

$$\Sigma_\beta = \begin{bmatrix} 2.1768 \times 10^{-11} & -1.0047 \times 10^{-10} \\ -1.0047 \times 10^{-10} & 6.6978 \times 10^{-10} \end{bmatrix}, \quad (48.36)$$

respectively. The mean estimate of the standard deviation of the error term ϵ_β is $\sigma_{\epsilon_\beta} = 2.1653 \times 10^{-5}$. The fitting and prediction results are shown in Fig. 48.13.

Probabilistic LCF Life Prediction Incorporating Surface Uncertainty

A probabilistic low-cycle fatigue (LCF) model considering the surface uncertainty is proposed. The basic idea is to incorporate the surface state parameter ρ into the classical Manson-Coffin model in Eq. (48.24), so that the uncertainty

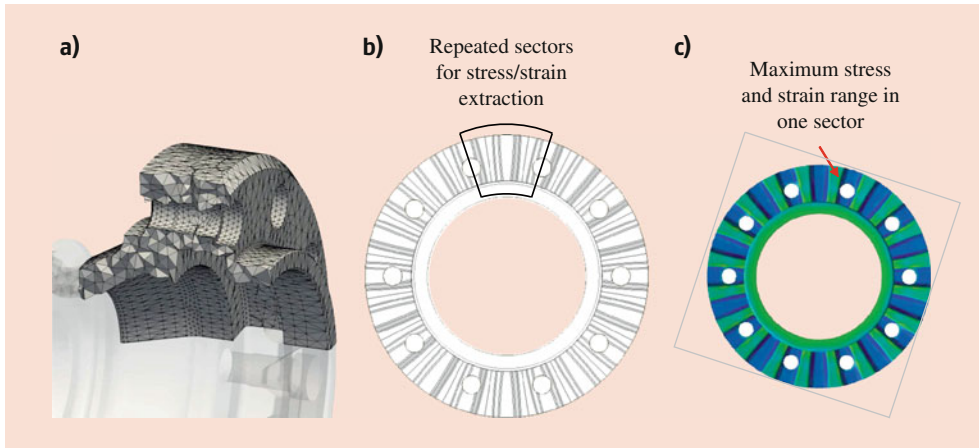


Fig. 48.12 The maximum stress/strain range at critical location in one of the ten repeated sectors. (a) Partial view of the finite element mesh, (b) the sketch of the Curvic coupling and the repeated sector, and (c) the

critical location in one sector where the maximum stress/strain range occurs in the representative load cycle in Table 48.1

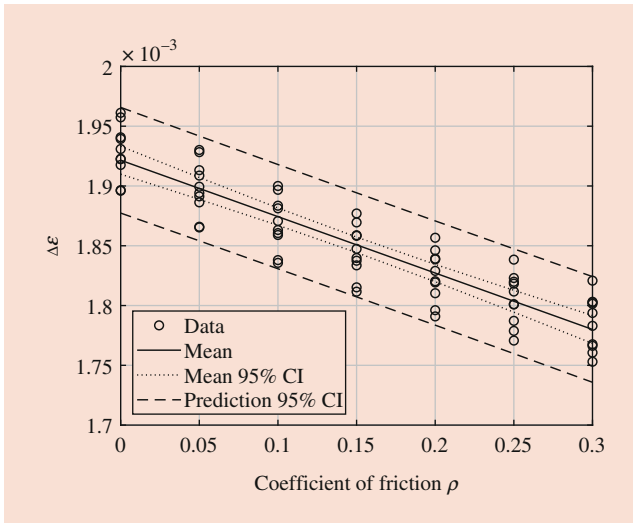


Fig. 48.13 The maximum strain range and the linear model fitting and prediction results using Eq. (48.34)

from the surface state can propagate through the model to the final fatigue life prediction. The low-cycle fatigue testing data on CrMoV material reported in Ref. [38] are used to estimate the model parameters with Eq. (48.24), and the mean and covariance matrix of the parameter $b = [b_0, b_1]$ are

$$\mu_b = [5.358, -1.332], \quad (48.37)$$

and

$$\Sigma_b = \begin{bmatrix} 9.729 \times 10^{-3} & 6.394 \times 10^{-3} \\ 6.394 \times 10^{-3} & 6.718 \times 10^{-3} \end{bmatrix}, \quad (48.38)$$

respectively. The standard deviation of the error term is estimated as $\sigma_{e_b} = 0.148$. The low-cycle fatigue data and

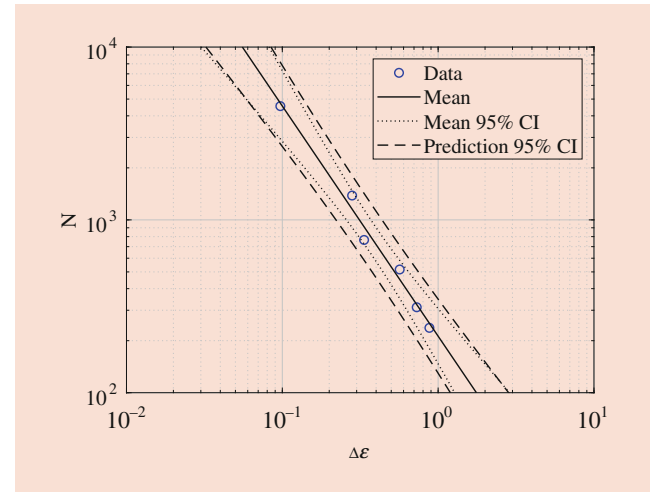


Fig. 48.14 The fitting and prediction results on CrMoV low-cycle fatigue testing data (source [38]) using Manson-Coffin model

Manson-Coffin model fitting and predictions are shown in Fig. 48.14.

Substitute Eq. (48.34) into Eq. (48.24) to obtain the surface-dependent low-cycle fatigue life model:

$$\ln N = b_0 + b_1 \cdot \ln(\beta_0 + \beta_1 \cdot \rho + \epsilon_\beta) + e_b, \quad (48.39)$$

where ϵ_β and e_b are error terms due to surface uncertainty and material uncertainty, respectively. Given the PDFs of the six parameters $\theta = (b_0, b_1, \beta_0, \beta_1, \sigma_{\epsilon_\beta}, \sigma_{e_b})$, the probabilistic fatigue life prediction can be made using simulation-based methods such as Monte Carlo (MC) simulations and its variants. The variation of static friction of coefficient for steel and steel in greasy condition is between 0.11 and 0.23 [39]. Without loss of generality, the uniform distribution in the range of [0.10.25] is used for the friction of coefficient

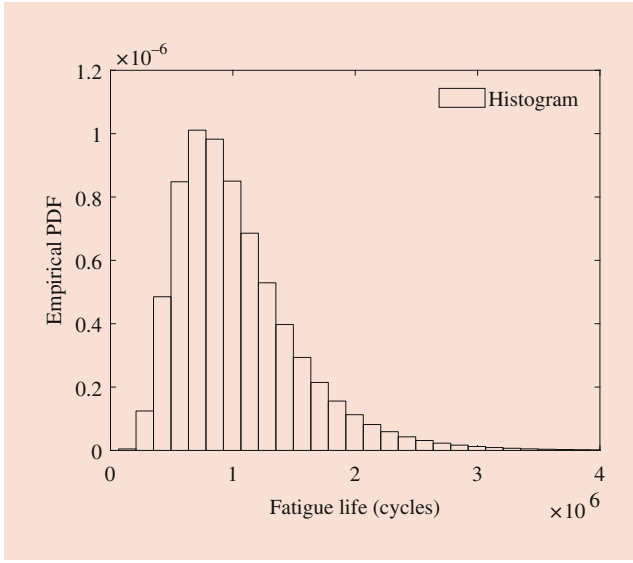


Fig. 48.15 The empirical fatigue life distribution obtained using 1×10^8 MC simulations

parameter ρ . The parameter $b = [b_0, b_1]$ follows a bivariate normal distribution with a mean vector of Eq. (48.37) and a covariance matrix of Eq. (48.38), respectively. The parameter $\beta = [\beta_0, \beta_1]$ follows a bivariate normal distribution with a mean vector and a covariance matrix given in Eq. (48.35) and Eq. (48.36), respectively. The two Gaussian error terms ϵ and e have zero mean and standard deviations of $\sigma_{\epsilon_\beta} = 2.1653 \times 10^{-5}$ and $\sigma_{e_b} = 0.148$, respectively.

A total number of 1×10^8 MC samples of $\theta = [b_0, b_1, \beta_0, \beta_1, \epsilon_\beta, e_b]$ are drawn according the above distribution parameters. For each of the samples, a deterministic fatigue life can be obtained using Eq. (48.34). With all the samples, the empirical distribution based on the resulting 1×10^8 fatigue lives can be estimated. The results are shown in Fig. 48.15. The quantile of the fatigue life prediction can be used to estimate the failure probability. For example, the quantile point value $N_{q,MC}$ with $q = 10^{-6}$ is the estimate for probability $\Pr(N > N_q) = 10^{-6}$. In this case, the estimated fatigue life with $q = 10^{-6}$ is found to be 1.03×10^5 cycles.

Time-Dependent Reliability Under Surface and Material Uncertainty

Consider a simplified loading profile of the rotor consisted of the nominal speed and the overspeed. For simplicity, assume the overload increase the maximum strain range at critical locations by 12% due to overspeed, overheating, etc., and the overload corresponds to a ratio of 10% in the designed lifetime in terms of cycles. The load block is shown in Fig. 48.16. This case can be modeled using the aforementioned block-type of loading. In this case $f_j = (0.1, 0.9)$ and $\Delta\epsilon_j = (1.12\Delta\epsilon, \Delta\epsilon)$. Here the term $\Delta\epsilon$ is given by Eq. (48.34), and

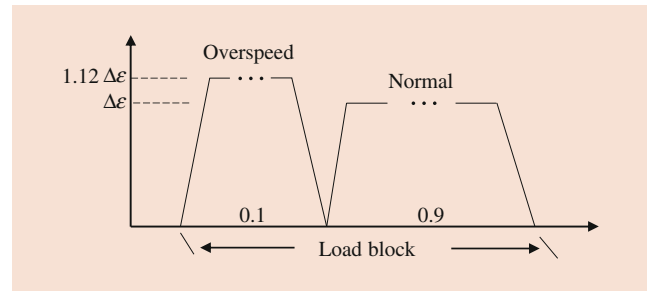


Fig. 48.16 The load block consisting 10% of overspeed and 90% of normal speed

N_j is given by Eq. (48.39). Using the linear damage rule in Eq. (48.25), the total damage accumulated after t cycles is

$$\Delta d(\theta, t) = t \cdot \left[\sum_{j=1}^2 \frac{f_j}{N_j} \right]. \quad (48.40)$$

The failure event occurs when the consumed accumulation exceeds 1 or more conservatively, a number between 0 and 1. The limit state function which defines the hypersurface separating the failure domain and normal domain can be expressed as

$$h(\theta, t) = \Delta d(\theta, t) - 1. \quad (48.41)$$

A total of 1×10^8 MC samples of $\theta = [b_0, b_1, \beta_0, \beta_1, \epsilon_\beta, e_b]$ are drawn. For each of the samples, the identity function Eq. (48.23) is evaluated with the limit state function of Eq. (48.41), and the time-dependent POF can be estimated using MC estimator in Eq. (48.22). The confidence interval of the MC estimator of POF can be obtained using Eq. (48.31). Results of the time-dependent reliability estimations and the time-dependent failure rate in terms of failures per cycle are shown in Fig. 48.17a and b, respectively. The risk requirement for life-limiting parts in aeroengine is about 10^{-9} failure event per flight cycle according to the FAA regulation [1]; the estimated life under the requirement is 1.3×10^5 cycles.

48.5 Summary

The chapter presents probabilistic models for fatigue reliability analysis using safe-life and damage tolerance methods. Probabilistic modeling is developed to formulate a general uncertainty quantification framework. Uncertainties from different sources are encoded into the distributions of model parameters. An inverse problem is set up for parameter estimation using the Bayesian method. A subsequent forward uncertainty propagation using the estimated parameter distributions can be used for probabilistic prediction. The safe-life and damage tolerance methods are incorporated into

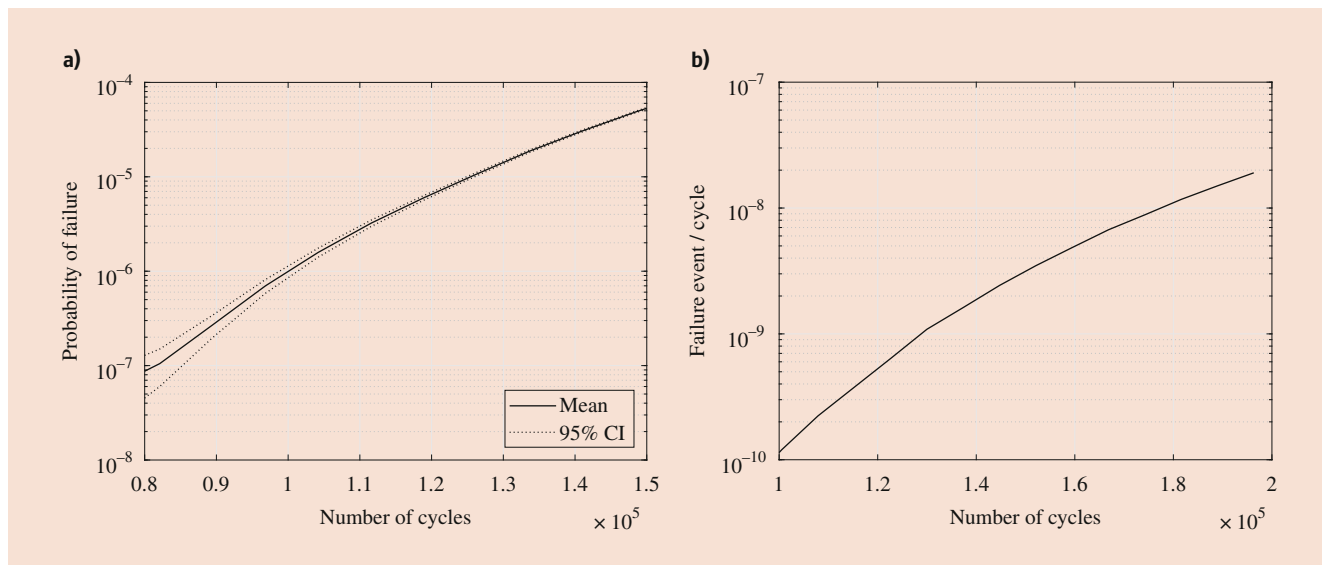


Fig. 48.17 (a) The time-dependent reliability in terms of probabilities of failure and (b) failure rate in terms of failures per cycle

the probabilistic framework for fatigue reliability analysis. The Monte Carlo estimators for reliability evaluations in terms of probability of failures are given. Two examples with realistic data are provided for demonstration. The difference between the median confidence interval and prediction confidence interval is signified using the examples.

The methods are applied to the reliability and low-cycle fatigue life prediction of a realistic Curvic coupling component in aeroengine. Contact surface uncertainty due to manufacturing and the inherent material uncertainty are considered. The coefficient of friction is used to represent different conditions of the surface state introduced by the manufacturing process. The strain and stress of Curvic couplings are evaluated using numerical models considering different contact surface coefficients of friction under working conditions. Based on the numerical results, a linear model is proposed to correlate the coefficient of friction and the resulting maximum principal strains at critical locations. Using the low-cycle fatigue model, a probabilistic life model incorporating the influence of surface roughness can be established to account for uncertainties from surface and material. Reliability and the fatigue life results are demonstrated.

References

1. Federal Aviation Administration, U.S. Department of Transportation: AC 33.14-1—Damage Tolerance for High Energy Turbine Engine Rotors. Federal Aviation Administration (2001)
2. Federal Aviation Administration, U.S. Department of Transportation: AC 33.70-2—Damage Tolerance of Hole Features in High-Energy Turbine. Federal Aviation Administration (2009)
3. Der Kiureghian, A., Ditlevsen, O.: Aleatory or epistemic? Does it matter? *Struct. Saf.* **31**(2), 105–112 (2009)
4. Guan, X., Jha, R., Liu, Y.: Model selection, updating, and averaging for probabilistic fatigue damage prognosis. *Struct. Saf.* **33**(3), 242–249 (2011)
5. Jeffreys, H.: An invariant form for the prior probability in estimation problems. *Proc. R. Soc. Lond. Series A* **186**, 453–461 (1946)
6. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., Teller, E.: Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**(6), 1087 (1953)
7. Hastings, W.: Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**(1), 97–109 (1970)
8. Kass, R., Raftery, A.: Bayes factors. *J. Am. Stat. Assoc.* **90**(430), 773–795 (1995)
9. Guan, X., He, J., Jha, R., Liu, Y.: An efficient analytical Bayesian method for reliability and system response updating based on Laplace and inverse first-order reliability computations. *Reliab. Eng. Syst. Saf.* **97**(1), 1–13 (2012)
10. Research Triangle Institute: Practical Reliability. Volume 4—Prediction CR-1129. Springfield (1968)
11. Basquin, O.: The exponential law of endurance tests. *Proc. Am. Soc. Test Mater.* **10**, 625–630 (1910)
12. Manson, S.: Behavior of materials under conditions of thermal stress Report 1170 Cleveland (1954)
13. Coffin, L.F. Jr.: A study of the effects of cyclic thermal stresses on a ductile metal. *Trans. Am. Soc. Mech. Eng.* **76**, 931–950 (1954)
14. American Society for Testing and Materials: ASTM E739-10(2015)—Standard practice for statistical analysis of linear or linearized stress-life (S-N) and strain-life (ϵ -N) fatigue data. ASTM International, West Conshohocken (2015)
15. Palmgren, A.: The service life of ball bearings (Translation of “Die Lebensdauer von Kugellagern”). *Z. Ver. Dtsch. Ing.* **68**(14), 339–341 (1924). NASA TT F-13460 (1971)
16. Miner, M.: Cumulative damage in fatigue. *J. Appl. Mech.* **12**, A159–A164 (1945)
17. Fatemi, A., Yang, L.: Cumulative fatigue damage and life prediction theories: a survey of the state of the art for homogeneous materials. *Int. J. Fatigue* **20**(1), 9–34 (1998)
18. Botev, Z.I., Grotowski, J.F., Kroese, D.P.: Kernel density estimation via diffusion. *Ann. Stat.* **38**(5), 2916–2957 (2010)
19. Glasserman, P.: Monte Carlo Methods in Financial Engineering, vol. 53. Springer, Berlin (2013)

20. Chu, F., Nakayama, M.K.: Confidence intervals for quantiles when applying variance-reduction techniques. *ACM Trans. Model. Comput. Simul.* **22**(2), 10 (2012)
21. Ministry of Defense: Design and Airworthiness Requirements for Service Aircraft Part 11: Engines 00-970 Part 11 UK (2018)
22. Harrison, G., Winstone, M.: Modelling and lifing of structural materials for future aeroengine components. *Adv. Perform. Mater.* **3**(3-4), 263–278 (1996)
23. Cláudio, R., Branco, C., Gomes, E., Byrne, J., Harrison, G., Winstone, M.: Fatigue life prediction and failure analysis of a gas turbine disc using the finite-element method. *Fatigue Fract. Eng. Mater. Struct.* **27**(9), 849–860 (2004)
24. Department of Defense: Engine Structural Integrity Program (ENSIP), MIL-HDBK-1783B w/CHANGE 2 edn. (WRIGHT-PATTERSON AFB OH, 2004)
25. Paris, P.C.: The fracture mechanics approach to fatigue. In: Burke, J.J., Reed, N.L., Weiss, V. (eds.) *Fatigue—An Interdisciplinary Approach*, pp. 107–132. Syracuse Univ. Press (1964)
26. Wheeler, O.: Spectrum loading and crack growth. *J. Basic Eng. Trans. ASME* **94**, 181–186 (1972)
27. Paris, P.C., Tada, H., Donald, J.K.: Service load fatigue damage—a historical perspective. *Int. J. Fatigue* **21**, S35–S46 (1999)
28. Borrego, L., Ferreira, J., Da Cruz, J.P., Costa, J.: Evaluation of overload effects on fatigue crack growth and closure. *Eng. Fract. Mech.* **70**(11), 1379–1397 (2003)
29. Forman, R., Kearney, V., Engle, R.: Numerical analysis of crack propagation in cyclic-loaded structures. *J. Basic Eng.* **89**(3), 459–464 (1967)
30. Walker, K.: The effect of stress ratio during crack propagation and fatigue for 2024-T3 and 7075-T6 aluminum. In: Rosenfeld M. (ed.) *Effects of Environment and Complex Load History on Fatigue Life*, pp. 1–14. ASTM International, West Conshohocken (1970)
31. Kujawski, D.: A fatigue crack driving force parameter with load ratio effects. *Int. J. Fatigue* **23**, 239–246 (2001)
32. Virkler, D., Hillberry, B., Goel, P.: The statistical nature of fatigue crack propagation. *J. Eng. Mater. Technol.* **101**, 148–153 (1979)
33. International Organization for Standardization: *Metallic Materials: Fatigue Testing: Fatigue Crack Growth Method. ISO 12108. ISO* (2002)
34. Guan, X., Zhang, J., Zhou, S., Russellkorde, E.M., Abbasi, W.: Probabilistic modeling and sizing of embedded flaws in ultrasonic non-destructive inspections for fatigue damage prognostics and structural integrity assessment. *NDT E Int.* **61**, 1–9 (2014)
35. Moeckel, C.W.: Probabilistic turbine blade thermal analysis of manufacturing variability and toleranced designs Ph.D. Thesis (2006)
36. Benardos, P., Vosniakos, G.C.: Prediction of surface roughness in CNC face milling using neural networks and Taguchi's design of experiments. *Robot. Comput. Integr. Manuf.* **18**(5–6), 343–354 (2002)
37. Cohen, D., Kligerman, Y., Etsion, I.: The effect of surface roughness on static friction and junction growth of an elastic-plastic spherical contact. *J. Tribol.* **131**(2), 021404 (2009)
38. Singh, V., Raju, P., Nambodhiri, T., Rao, P.R.: Low-cycle fatigue behaviour of a low-alloy high-strength steel. *Int. J. Fatigue* **12**(4), 289–292 (1990)
39. Gray, D.E. (ed.): *American Institute of Physics Handbook*. McGraw-Hill, New York (1972)



Xuefei Guan received his PhD from Clarkson University in the US in 2011. He worked at Siemens Corporate Research in Princeton NJ, US. His research area includes uncertainty quantification, structural reliability, and ultrasound nondestructive evaluations.



Jingjing He received her PhD from Clarkson University in the US in 2012. Her research area includes structural health monitoring, damage diagnosis and prognosis, and Lamb waves.



Application of Cognitive Architecture in Multi-Agent Financial Decision Support Systems

49

Marcin Hernes and Ngoc Thanh Nguyen

Contents

49.1	Introduction	981
49.2	Related Works	983
49.3	Recent Methods for Financial Decision Supporting	984
49.4	Cognitive Architectures	989
49.4.1	Basic Issues.....	989
49.4.2	Symbolic Architecture.....	990
49.4.3	Emergent Architectures.....	991
49.4.4	Hybrid Architecture.....	991
49.4.5	Characteristic of Cognitive Agents.....	991
49.5	The Architecture and Functionality of MAFDSS ...	996
49.6	Research Experiment	997
49.7	Conclusions	998
	References	999

using natural language processing to sentiment analysis for the consumer ambience and to buy/sell decision-making. The aim of this chapter is to present an approach to apply cognitive architecture in a multi-agent financial decision support system. On the basis of performed researches, it can be stated that cognitive architecture allows to increase the usability of the multi-agent system and consequently to improve the process of taking investment decisions. The ability to make automatic decisions is also of high importance here. For example, the cognitive agent can make real transactions (open/close short/long position) on the Forex market.

Keywords

Multi-agent systems · Cognitive technology · Fintech · Financial decisions · Artificial intelligence

Abstract

The multi-criteria character associated with making decisions entails the need for analysis and evaluation of a large amount of information and drawing conclusions on the basis of such information. Since the process is time-consuming and practically impossible to be performed by a decision-maker in real time, it is necessary to use computer decision support systems, including multi-agent systems. To support financial decision-making, the cognitive technologies can be used. These technologies support

49.1 Introduction

The financial market is characterized by high variability of operating conditions [1]. Therefore, making financial decisions is a continuous process, associated with multi-criteria decisions nature. Subsequent decision situations occur chronologically in near real time and are always associated with risk.

The multi-criteria character associated with making decisions entails the need for analysis and evaluation of a large amount of information and drawing conclusions on the basis of such information. Since the process is time-consuming and practically impossible to be performed by a decision-maker (a human) in real time, it is necessary to use computer decision support systems, including multi-agent systems. These systems enable an automatic and quick way of finding information characterized by the appropriate value and drawing conclusions based on such data [2]. As a rule, each agent in the system uses a different method for financial

M. Hernes (✉)
Faculty of Management, Wroclaw University of Economics and Business, Wroclaw, Poland
e-mail: marcin.hernes@ue.wroc.pl

N. T. Nguyen (✉)
Faculty of Information and Communication Technology, University of Science and Technology, Wroclaw, Poland
e-mail: ngoc-thanh.nguyen@pwr.edu.pl

decision support, as well as can analyze different, often heterogeneous, data sources. A certain number of agents can, for example, make decisions using technical analysis, based on various types of indicators, which involves processing mainly structured data. Increasingly, however, agents make decisions using fundamental or behavioral analysis, which involves processing of expert or investor opinions. These opinions can be found, inter alia, on websites of brokerage houses, banks, or on social networks/blogs about financial matters. They are written mainly in a natural language, which is associated with the problem of processing unstructured data.

Parallely, there are observed changes in the financial industry related to the emergence of technology startup enterprises, which were previously outside of the financial sector. They are called “*Financial Technology*” or “*Fintech*” enterprises [3]. Mondal and Singh [3] state that key research issues for realizing innovations in the financial services include:

- Modeling and integrating large-scale and complex data from disparate and heterogeneous sources
- Putting a context around the data for context-aware reasoning and analytics purposes
- Cognitive analytics and semantic understanding and interpretation on the financial data
- Effective heuristics for detecting fraud
- Intelligent approaches for identifying cases of “creative accounting”
- Techniques for working with huge amounts of uncertain data from different modalities and identifying cross-connections and correlations

Dapp [4], in turn, states that current challenges in Fintech are related to using intelligent, cognitive, and self-learning technologies to support financial decisions-making processes. The authors have observed that new Fintech solutions are easily adopted by customers.

Cognitive technologies serve as tools for problem-solving that integrate such techniques, as machine learning, big data, data mining, computer vision systems, robotics, and natural language processing. Cognitive technologies can be used to support financial decision-making [5]. These technologies support using natural language processing (NLP) to sentiment analysis for the consumer ambience and to buy/sell decision-making [5]. Schumaker et al. [6] state that in order to support taking financial decisions, numerical and textual data should be processed parallely. For example, as a supplement of technical analysis (based on quotation numerical values) also the news (for example, from Tweeter) related to particular quotations should be automatically analyzed [7]. Jung et al. [8] and Alt et al. [9] address this problem

by applying a combined approach based on the technology acceptance theory and cognitive technology. Nowadays, machine learning, including deep learning, is mainly used to support taking financial decisions [10, 11]. They are useful for analyzing large volumes of numerical data. Thus, they are called cognitive tools in the explicit sense. However, a very big challenge is that they still are not enough for a computer system to learn a natural language and to understand the context of phenomena occurring in the environment as quickly and accurately as a several-year-old child [12, 13].

To resolve this problem, cognitive architectures, in the implicit sense (based on cognitive cycle), can be used. Generally speaking, the process of learning by humans is executed in the form of a cognitive cycle, from perception to purposeful actions. Many different artificial intelligence techniques are needed to support the cognitive cycle. Cognitive technologies must integrate them in a cognitive cycle of perception, learning, reasoning, and actions. This cycle is the key to designing modern intelligent systems [13]. The cognitive cycle consists of the following main stages [13]:

- Induction of new observations and generalizations of knowledge
- Abduction – the process of revising beliefs to modify theories
- Deduction, i.e., prediction
- Actions whose effects may confirm or refute the results of previous stages

The process of learning is also very important – chunks of knowledge are organized into theories – a set of consistent beliefs. The value of theory is high, if predictions based on this theory lead to successful actions. Learning is a permanent process of transforming data to theories. The cognitive cycle is a framework for accommodating multiple components of any kind [12, 13].

The aim of this chapter is to present an approach to apply cognitive architecture in a multi-agent financial decision support system (MAFDSS). This system is built using the framework Learning Intelligent Distribution Agent (LIDA) [14]. In this chapter, we consider the cognitive technology in the implicit sense (technologies based on cognitive sense consists of different artificial intelligence techniques).

The first part of the chapter presents the analysis of related works in the considered field. Issues related to cognitive technologies’ characteristics are presented in the second part. Next, the architecture of MAFDSS and functionalities of text analysis-based buy/sell decision agent are presented. The last part of the chapter presents the method of conducting and the results of a research experiment aimed at verifying the selected cognitive agents.

49.2 Related Works

There are different approaches to supporting financial decision-making process using multi-agent systems. Luo et al. [15] presents a multi-agent system for the stock trading framework, which consists of the following types of agents: the interface agent; the coordinator agent; the profiler agent; the monitoring agent; the communication agent; the risk management agent; the decision-making agent; the technical analysis agent; and the fundamental analysis agent. Such framework provides high performance of systems, but the authors draw attention to one main weakness – the coordinator agent is the critical component of this framework. The reason for this is that if this agent fails the whole system cannot work correctly.

Hafezi et al. [16] present a bat-neural network multi-agent system (BNNMAS), which aims to predict the stock prices. This framework uses four-layer genetic algorithm neural networks (GANN) and the generalized regression neural network (GRNN). Tan and Lim [17], in turn, introduced a hybrid intelligent system based on case-based reasoning (CBR) and the fuzzy ARTMAP (FAM) neural network model. This system supports investment decision-making process in manufacturing enterprises. The system stores cases of past investment projects in a database, characterized by a set of attributes indicated by human experts. The FAM network compares the attributes of a new project with attributes from past cases. Some similar projects are retrieved and adapted. The information from these projects can be used as an input to developing new investment projects.

Gottschlich and Hinz [18] present a system for supporting financial decision-making process based on collective wisdom. It allows for taking into consideration the crowd's recommendations for supporting investment decisions.

A numerical Pareto optimization method based on an evolutionary strategy has been presented by Xia [19] and Roosen et al. [20]. Barbosa and Belo [21] present a multi-agent approach based on a set of autonomous currency trading agents, using classification and regression models, a case-based reasoning system, and an expert system. Each agent supports a decision for different currency pairs. In order to decrease the level of risk the agents share monetary resources which have been used. Following such an approach to investments allows for achieving better results.

Some decision support systems related to financial crisis based on the investors' psychological behavior and rational reasoning are presented by Said et al. [22]. They focus on three main biases: overconfidence, loss aversion, and mimetic behavior. Authors state that for analyzing the financial crises, the interaction between rational and irrational behavior and the investor's psychology must be taken into consideration.

Chen et al. [23] present a hybrid multi-agent model developed based on mathematics and economics models. Here, more qualitative features are considered. The authors state that economic features, which cannot be modeled mathematically, can be difficult to be verified experimentally. Serrano [10] presents a random neural network in a deep learning cluster structure with learning based on a genetic algorithm, where information is transmitted in the combination of genes rather than the genes themselves. This approach has been applied and validated in Fintech; O'Hare and Davy present a solution based on classification of text opinions about financial markets using machine learning techniques. They use a multinomial naive Bayes (MNB) classifier and a support vector machine (SVM), for opinions' sentiment polarity classification [24]. Wang [25], in turn, presents an approach for supporting taking financial decisions based on the Matlab neural network time series tool. Sohangir et al. [11] applied several neural network models such as long short-term memory, doc2vec, and convolutional neural networks, to stock market opinions' analysis posted in StockTwits. Also, Li et al. [26] use convolution neural network (CNN) and two recurrent neural networks (RNNs) – the simple recurrent network (SRN) with traditional recurrent units and a long short-term memory (LSTM) with gating mechanisms, for analyzing opinions of investors. The machine learning approach is also used by Chan and Chong [27]. They develop two classifiers:

1. Making prediction of the right chunking point that lies between two adjacent phrases
2. Figuring out the appropriate syntactic structure for the chunks, such as noun phrases

Westerhoff [28] describes a system in which agents were divided into two groups – agents of the first group make decisions on the basis of methods of fundamental analysis, while agents of the second group make decisions on the basis of technical analysis. Paper [29] presents a multi-agent system that supports investing in the Forex exchange market and a method for assessing investment strategies of selected agents. Korczak et al. [30] conclude that different AI methods are used for supporting financial decision-making processes, including:

- Strong AI
- Deep learning
- Robotics
- Advanced brain-machine interfaces
- Trisynaptic models
- Circuit of the hippocampus
- Massive parallel processing
- Feedforward as well as feedback inhibition

In the process of document analysis, implemented also in multi-agent financial decision support systems, the following methods are used:

- Information retrieval
- Information extraction
- Text exploration
- Natural language processing [31]

The main purpose of information retrieval is to find an answer to a user's question among a collection of documents. Information extraction consists in identifying instances of a predefined class of events, their connections, and occurrences in written documents in the natural language [32]. The aim of text exploration is to learn the hidden information in the text using methods adapted to a large number of text data [33]. The natural language processing contains mechanisms that attempt to "understand" the context of the text. These methods do not include the term similarity values, but the following categories of text analysis are carried out [33]:

- The shallow analysis. It is referred to the analysis of the text, the effect of which is incomplete in relation to the deep analysis. Usually, the limitation is the recognition of nonrecursive or restricted recursion patterns that can be recognized with a high degree of certainty.
- Deep text analysis is the process of computer linguistic analysis of all possible interpretations and grammatical relations occurring in the natural text. Such a full analysis can be very complex.

In the process of text document analysis, semantic methods of knowledge representation, including semantic networks, are often used. Thanks to their application, a broadly understood knowledge representation is possible, in which it is important to draw attention to the interrelations between objects.

A very important problem is also the sentiment analysis related to other, than English, languages. Many tools for texts analysis in English are insufficient for other languages and it is needed to develop new, or to adjust existing tools to a given language, which is also a very important research and practical problem. For example, Ahmad et al. [7] develop the grid-based sentiment analysis method for Arabic and Chinese languages.

To sum up, in multi-agent systems supporting financial decision-making, the following methods are mainly used: technical analysis, fundamental analysis, and behavioral analysis. They are based on statistical methods and traditional data mining methods, such as regression analysis and decision trees and on training methods. Hassouna et al. [34] have showed limitations of statistical methods and traditional methods of data mining, such as:

- They do not explain the reasons for obtaining the value of retention and the relationship between factors that affect these values.
- Although they allow to define connections between variables and customer behavior, they do not allow to identify causal relationships between these variables. Decision trees can only be used in some cases to infer causation.
- Data mining models have a relatively short validity period. The application market, including mobile applications, faces new technologies every day. As a result, historical data becomes less valuable for prognosis.
- The level of analysis in data mining models reduces the ability to capture the heterogeneity of customer behavior.

In order to circumvent the disadvantages of traditional data mining techniques, the machine learning, including deep learning, models are currently used. However, these models also have limitations resulting from the following issues:

- Emergent character of ML – processing only numeric data. Many problems of the real world often also need symbolic data representation.
- The difficulties which use the hierarchical data (e.g., trees, thesauruses, and relations 1:M in databases) as input data model.
- The difficulties related to incomplete, contradictory data representation (e.g., "null" values).

In order to reduce the number of disadvantages of traditional data mining and deep learning methods, cognitive agents can be used in the multi-agent system supporting taking financial decisions. They perform cognitive and decision-making functions, which take place in the human brain, thanks to which they are able to analyze the real significance of observed phenomena and processes taking place, among others, on financial markets [12].

49.3 Recent Methods for Financial Decision Supporting

The methods for supporting financial decisions are generally divided into fundamental analysis, technical analysis, and behavioral analysis.

Using **fundamental analysis**, the investor is only interested in the issuer of the security and the economic environment in which they operate. By examining their financial condition, strengths, weaknesses, and development prospects, the investor answers the question whether it is worth investing in the issuer's securities [35]. A comprehensive examination of the issuer's condition is accompanied by an analysis of their macroeconomic environment, which allows to identify the entity which has the best opportunities

for future profit growth. Fundamental analysis is a form of capital market valuation involving the assessment of processes taking place in the macroeconomic environment of the stock exchange.

When choosing the most attractive companies, supporters of fundamental analysis refer, among others, to the following [35]:

- Audits of periodic statements and annual balance sheets of companies
- Analysis of the company's future profit forecasts
- Analysis of profits generated in a given period as well as incurred losses
- Examination of the development strategy chosen by the company

The investment selection process also applies to [36] the following:

- The general economic condition of the country
- The monetary policy implemented by the government and the central bank
- The analysis of the industry in which the company operates
- Strengths and weaknesses, opportunities, and risks which the company faces

It follows from the above that the fundamental analysis is used to determine the profitability of investments in company shares based on a comprehensive assessment of the company from the past. The conclusions are reached after analyzing the results of a minimum period of 3 years of the company's activity. To achieve a satisfactory result, the time projection of analyzing the company's results should be extended to 5 years.

The sense of fundamental analysis is not to identify good companies, but to select companies that are better than anyone can suppose or judge. Similarly, companies negatively perceived by the market can be excellent investment opportunities, if, of course, they are not as bad as it seems. Therefore, it is not enough to conduct a good fundamental analysis; one can only earn money when the conducted analysis is of higher quality than the analysis of the competition [36].

Opponents of fundamental analysis often question the possibility of including a large number of different types of risk in the company's valuation, such as interest rate risk related to the change in basic market interest rates, market currency risk, purchasing power risk – also known as inflation risk, political risk related to legislative and tax changes, etc. Fundamental analysis is not an ideal method, as it possesses various disadvantages which affect the results of investment decisions. Here are the most important of them [37]:

- The assumption that the company whose profits have risen so far will continue to strengthen; the profits of companies are not growing because they did so in the recent past, but because the companies are well managed, have better technology, and are well promoted.
- The assumption that the growth rate will be maintained at its current level.
- There are no fluctuations in economic life; expansions and recessions occur in the capitalist economy, the effects of which are felt by almost all enterprises.
- Failure to take account of the fact that market prices precede fundamental knowledge.

One of the major disadvantages of fundamental analysis is the assumption that the current trends disclosed in the financial reports will continue. In real economic life trends change, which makes forecasting much more difficult. The fundamental analysis consists of four stages: analysis of the macroeconomic environment, financial analysis, analysis of financial statements, and analysis of indicators. Let us, therefore, discuss their brief characteristics.

The initial stage of the fundamental analysis is to determine the state of the economic environment and its impact on investment decisions, i.e., the analysis of the macroeconomic environment. The concept of growth is central to this type of stock analysis, as the increase in profits and cash flow is considered the basic condition for the increase in dividends and share prices.

Macroeconomic analysis provides data used in industry and enterprise analysis, and allows forecasting interest rate levels and long-term trends in price and profit ratios. Analysts try to identify sectors of the economy that offer higher-than-average profit opportunities, and then relate this information to specific companies.

Economic cycles and related trends are crucial for the investor. The cyclicity of the stock market has been around since 1961 and has become a major disincentive to invest in the late 1970s and early 1980s [38].

Analysts who use fundamental analysis attach great importance to profits achieved by enterprises as well as to the analysis of the macroeconomic environment.

The macroeconomic analysis assesses the overall attractiveness of investing in a given stock market. This attractiveness depends on the economic and socioeconomic situation, as well as on the economic and monetary policy of the country in which the investor wants to invest, with a particular emphasis on the investment risk in the given country. Subsequently, a sectoral analysis is performed, which assesses the attractiveness of investing in companies belonging to a given sector of the economy. After choosing a sector, it is the turn of the analysis of companies belonging to a given sector. At the beginning, a situational analysis of the company is made

(against other companies in the sector), where nonfinancial aspects are assessed, e.g. [37]:

- Quality of management staff
- Modernity of production
- Company strategy
- Conducted marketing, etc.

It is clear that this stage of fundamental analysis is not easy for practical application by an individual novice investor. Such an investor should be recommended a “short-cut.” It involves studying ready-made analyses which are published in financial magazines or in financial columns of newspapers.

When choosing the most attractive companies, a special role in fundamental analysis is played by financial analysis, which is divided into three basic groups of investment techniques [38]:

1. Examination of the vertical and horizontal structure of the company’s balance sheet. This analysis includes determining the regularity of financing investment activities and the method of financing fixed assets, the method and regularity of financing current stocks, the company’s payment ability, and the ratio of receivables to liabilities. Therefore, the balance sheet division determines the dynamics and development opportunities of the company, while the horizontal layout checks its quality and market value.
2. Analysis of changes in the company’s foreign capital. The amount of foreign capital illustrates the degree of financial security that an entity possesses. It also informs about the amount of capital that will remain inside the entity after paying off its current liabilities. If the foreign capital is at a high level, this indicates a high investment risk and carries a possible danger to the company in the event of nonpayment of its obligations. The optimal situation is when foreign capital is at a relatively low level, which makes it possible to state that the given entity is in good financial condition and it is unlikely that its position among the competition will be violated.
3. Determining the company’s cash flow. This is the sum of profit and depreciation for individual tasks that have already been completed. This factor provides information on the sources and purpose of cash, as well as the type and method of financing the investment.

Technical analysis is not interested in the security issuer it deals with. Using this method, all that is needed is the current exchange rate and turnover, past data, and a reference point to calculate the indicators. It is not necessary to know the name of the security that is being analyzed in order to determine the optimal buying and selling times [39].

Technical analysis is based on three basic premises [40]:

1. The market discounts everything. Supporters of technical analysis claim that all factors that affect the price are already included in it. This is due to the belief that price behavior reflects changes in demand and supply relationships. In other words, the technical analyst assumes that one does not need to study the factors affecting the price of the instrument or investigate the reasons for the decreases or increases. The market knows everything and it shapes the price, which is why it is important to study charts and indicators for technical analysis to effectively forecast market behavior. This does not mean that technical analysts reject the claims that economic conditions are the cause of trends on the stock market. They simply believe that the market is easier to understand and predict by analyzing the record of its behavior, i.e., the charts.
2. Prices are subject to trends. When drawing price charts, the technical analyst tries to find a trend in them, i.e., the direction in which prices are moving. Recognizing the trend in its early phase allows to make a transaction that should bring profits (buy when the upward trend is created, sell when the downward trend appears). Analysts assume that the trend is more likely to continue than to reverse.
3. History repeats itself. The study of charts allows to find repetitive patterns (formations) by which prices move. This is due to the repetition of human behavior in certain situations. Knowing the most common patterns (formations), analysts try to find them in current quotations and on this basis forecast the future.

To sum up, the first premise claims that the market fully discounts all information that may affect the stock price, which means that testing prices alone is a self-sufficient approach. The second premise concerns the trend, one of the basic concepts of technical analysis. It claims that courses move in trends, and the task of technical analysis is to capture their changes. The third premise concerns market psychology and assumes that under similar conditions to the past, the market situation will be similar.

Therefore, a technical analyst is a person who studies the effects of market behavior (prices) and searches trends and formations that will help predict market behavior on historical charts. In their work, analysts support themselves with, among others, indicators which help them better understand price movements [41].

Thanks to this, it is possible to find the answer to three main questions that every investor asks [38]:

1. What to buy or sell?
2. When to buy?
3. When to sell?

Table 49.1 Technical analysis indicators [38, 41]

Tracking indicators trend	Oscillators	Mood indicators
They work best when prices are “in motion” but their signals are incorrect in the horizontal trend; they are lagging behind the trend, changing direction when reversed	They capture turning points in flat markets; however, their signals are premature when a new upward or downward trend begins; they are indicators that are concurrent with or even preceding a trend	Allow to study the “psychology of the masses”; these indicators may be leading or concurrent with the trend
Moving averages MACD MACD – histogram OBV – On Balance Volume A/D – accumulation/distribution	Stochastic ROC – rate of change Momentum RSI – Relative Strength Index Force Index CCI – Commodity Channel Index	New High-New Low Index Put-call ratio Bullish Consensus Commitments of Trade Advance/Decline Index Traders

Technical analysis, like fundamental analysis, is used to determine the right time to buy and sell shares. To make the investment as profitable as possible, a variety of tools are used in technical analysis in order to study phenomena occurring on the stock market. However, unlike the fundamental analysis, the whole process of assessing the amount of future investment profits proceeds in a slightly different way. In technical analysis, investors are guided in the stock market game mainly by historical data, which do not have their roots in reports, balance sheets, or profit and loss statements, but in data from the direct economic environment of a given company and what happens directly on capital market. These data are used by investors to construct stock market indicators, helpful in assessing the attractiveness of shares of selected companies. The first stock market indicators were created at the end of the nineteenth century, but their full boom can be considered only the period of the great bull market in the USA from the 1920s. Since then, their popularity has been established and continues to this day. The breakdown of indicators is presented in Table 49.1.

The behavioral trend is not intended to provide guidelines for rational behavior, as standard economic theory does, but to describe and explain the mechanisms that control people’s real decisions, mechanisms which stem from psychological and institutional factors [42]. Numerous theories serve this purpose, several of which are described below. The common basis for each of these theories is the assumption that people do not have perfect information or unlimited possibilities to process it; in the real world, information is incomplete, unclear, complicated, there may occur problems with its interpretation, and for practical reasons we are often unable to compare all possible selection options in each of the relevant aspects. For the abovementioned reasons, it cannot be said that people take optimal actions from the point of view of material rationality. Instead of striving to achieve maximum utility or profit, we are content only with their satisfactory level (satisfying behavior). In behavioral terms, the utility of the decision-maker is not treated as a permanent, coherent, and one-dimensional category [43]. On the contrary, several types are distinguished, among others:

- Decision-making utility, most often found in traditional economic approaches and referring to preferences that are disclosed by the buyer at the time of purchase
- Experienced utility, which relates to the perceived level of satisfaction

The literature mentions two research directions related to behavioral finance: analysis of market behavior on a macroeconomic scale, especially in the context of numerous market anomalies, and analysis of individual investors’ behavior in terms of systematic cognitive errors made by them in investment decisions.

Behavioral analysis usually employs the following [44]:

- Descriptive theory
- Predisposition effect
- Emotion changeability
- Cognitive distortions

Descriptive theory is the descriptive (i.e., based on the results of experiments) theory of decision-making in conditions of uncertainty, and it can be considered as the greatest contribution of psychological sciences to economics and finance. It is a model that somehow filled the gap between the idealistic theory of the expected utility of von Neumann and Morgenstern, and reality. The central assumption of the theory of perspective is that it is not the absolute level of wealth, but changes in its level (profit vs. loss) that are the carriers of perceived value. Another important assumption of this theory is decreasing sensitivity to losses. The last important feature of the human psyche affecting the shape of the function of value is the loss aversion, which is that losses hurt more than profits please, despite the fact that they are of the same absolute size. This means that the value function is steeper in the area of losses than in the area of profits [42].

The predisposition effect is manifested in the fact that shares in companies whose prices have risen are sold much faster by individual investors compared to shares in companies that have lost value. This is contrary to the classic theory of finance, according to which in making investment

decisions investors should be guided by expectations about future stock prices and not their price in the past. Moreover, the predisposition effect also negates two central assumptions of the portfolio theory: the tendency to diversify one's own investment portfolio and the investors' desire to sell shares in order to increase their own liquidity. In both situations, investors should sell those shares which possess the lowest expected returns. In addition, it would be advisable to sell shares of loss-making companies quickly, as this is the optimal solution from the tax point of view. Meanwhile, according to the predisposition effect, investors do the opposite, i.e., they are more willing to sell shares that have a higher expected rate of return [42].

Investing in the stock market is frequently associated with violent and variable emotions, which often have a significant impact on decisions taken, causing significant deviations from rational action. Making decisions on the capital market takes place in conditions of considerable uncertainty, because investors never know how the market will behave in a month, in a week, or even in an hour. Psychologists have long proven that the behavior of an individual is strongly dependent on their emotional state (8). Additionally, cognitive errors and distortions, which are usually a derivative of the impact of emotions, have a significant relationship with the limited rationality of the individual (9). Analyzing the impact of emotions and the psyche of investors on making decisions on capital market, one can get a real picture of investor behavior on the stock market. One of the most important manifestations of the impact of emotions on the investment process is that investors give in to the feelings of regret. Regret often accompanies investors after making a decision whose consequences have proved unfavorable for them. The theory of regret formulated by David Bell (10) assumes that the feeling of regret associated with making the wrong investment decision is stronger than the possible feeling of pride and satisfaction associated with making the right choice [43].

What follows is hasty selling of rising shares in order to secure oneself against the future decline in their prices. In addition, the regret caused by the loss on investing in a particular company is experienced more intensely than in the case of a loss at the level of the entire stock exchange, because that loss can be attributed by investors not to their own mistake, but to unpredictable market forces. The regularities described here affect the way investment portfolios are constructed by individual investors who choose their portfolio so that some of them provide a reduction in fear of potential loss, and some give hope for future profits. This is, therefore, contrary to Markovitz's portfolio theory, as investment portfolios in a behavioral perspective are not a set of analyzed and suitably diversified assets, but rather a mosaic of little correlated elements that reflect the investors' emotions rather than a rational assessment of the market's situation [44].

In addition to emotions, investors often undergo various cognitive distortions when making investment decisions. One of them is overconfidence. Investors are often so convinced of their outstanding investment capabilities that they undertake increasingly risky transactions, which often ends in very severe losses. An important manifestation of excessive self-confidence is unrealistic optimism, visible in making mistakes in the planning process. Its important aspect is the inability to learn from one's mistakes – even if earlier forecasts were excessively optimistic, most people believe that the next time their choice will prove to be correct, which can also be fatal in consequences for their state of ownership. Another cognitive distortion is the so-called mental accounting, which involves a separate and selective analysis of the various attributes of financial decisions. This leads to the fact that investors are not able to close the loss positions and invest funds from their shares in new ventures that resulted in losses, because such shares are treated as a separate mental account. Instead, many of them hope to change the trend and make up for the losses in the future, which often deepens these losses and causes a further decrease in the value of the entire investment portfolio [42].

Finally, analyzing cognitive distortions in the process of investment behavior, one should mention heuristics. These are practical rules of thumb, based on experience or even colloquial knowledge and used to make decisions without a thorough comparison of all available options (13). If an algorithm can be treated as an exact recipe for solving a problem, heuristics is usually only useful as a tip that may not necessarily bring the correct solution to the problem. Heuristic distortions are quite common in the process of making investment decisions on the stock exchange, where information buzz prompts investors to make decisions based on heuristics [44].

The financial decision support is always related to the risk. Depending on the group of factors posing a threat to the purpose of a given investment in capital assets, the following types of risk can be distinguished [8]:

1. Political risk. It is associated with political events and decisions taken in the country. Political decisions that can affect the capital market (negatively or positively) include, for example, the result of the election leading to a change in the ruling political option [45].
2. Risk related to the situation on the stock exchange market. It is associated with continuous movements of securities prices. Stock market analyses lead to the conclusion that prices are changing in line with trends. Changing the direction of the trend from growing to decreasing and vice versa causes a large uncertainty in investment because it is impossible to clearly indicate the point at which the trend will change [8].

3. Market risk. It manifests itself in fluctuations in market prices. It is associated primarily with changes in raw material prices, supply prices, product selling prices on the market, etc. [8].
4. Liquidity risk. It is associated with the difficulty of liquidating investments. Liquidity should be understood as the time necessary to convert assets (e.g., securities) into cash. The longer this time, the greater the liquidity risk [46].
5. Inflation risk. It is associated with a change in inflation rate. In conditions of high inflation, this may mean that investment income does not cover the increase in the cost of maintaining it [47].
6. Interest rate risk. To a large extent, it is a consequence of inflation risk. The level of return depends on the interest rate offered by banks on the market. An increase in the interest rate results in the outflow of capital from the stock exchange and placing it on bank accounts. Bank deposits are much less risky than investing on the stock exchange market. The consequence of this situation is a decrease in the rate of return on investment and vice versa, lowering interest rates by banks leads to an increase in the attractiveness of investing in securities. A special case of interest rate risk is the risk of a price change, often referred to as holding period risk. It is characteristic of the bond market and is associated with a situation in which the bond holder intends to sell it before the date of buyout [48].
7. Exchange rate risk. It occurs when the financial instrument in which one invests is denominated in a currency other than that of the investor. In this situation, the rate of return expressed in two different currencies is not the same. Therefore, if the investor buys a financial instrument in a foreign currency and the exchange rate depreciates, this is a favorable situation for the investor. When the exchange rate appreciates, the investor will lose because the nominal return on investment will decrease [49].
8. Default risk. It occurs when the issuer of a financial instrument fails to comply with the terms of the contract. If the investor buys bonds issued, for example, by a listed company, and the latter does not pay interest or fails to timely redeem the bonds, the investor will incur a loss [50].
9. Risk of reinvesting. It is related to a situation in which an investor's income from holding a financial instrument is reinvested at a different interest rate (e.g., due to changes in interest rates on the market) than the rate of return on a given financial instrument [51].
10. Risk of redemption on demand. It is related to futures and bonds. For example, if the investor has issued an American option for sale, which can be exercised at any time, the investor will incur a loss if the request to exercise this option occurs when the value of the instrument for which the option was written decreases.
11. Risk of interchangeability. It occurs in the case of instruments that can be converted into other instruments (in Poland, an example of such financial instruments are convertible bonds issued by the State Treasury). The risk is that the exchange may occur in conditions that are unfavorable to the investor [52].
12. Financial risk. It is a state of uncertainty that appears when financing investments with foreign capital. The use of debt capital leads to additional financial charges associated with servicing the loan. The financial risk is strongly influenced by interest rate fluctuations, which are quite difficult to predict [53].
13. Banking risk. Banks are the second group, apart from enterprises, using derivatives to minimize the risk of their activities, i.e., minimizing banking risk. Banking risk is associated with an extremely important role played by banks in the national economy. This role boils down to the three most important areas of the bank's activity, which are [54]:
 - Participation in the creation of money
 - Participation in the social division of labor
 - Allocating and transforming funds

The risk issues should be taken into consideration regardless of the method used to support financial decisions.

The presented methods for financial decision supporting are often implemented in agents' architectures in multi-agent systems. On the basis of decisions and data generated by several agents (each agent uses a different decision support method), it is possible to provide more sophisticated investment strategies. For example, the strategy can be built on the basis of expert opinions (based on, for example, fundamental, technical, or behavioral analysis). Such strategies can be realized by cognitive architectures characterized in the next part of the chapter.

49.4 Cognitive Architectures

49.4.1 Basic Issues

The cognitive architecture is defined as an organized structure of processes and knowledge trying to model itself on cognitive dependencies [55]. The primary stimulant for the creation of cognitive architectures was helping humans in complex situations, where the computer could be characterized by a much faster reaction time in a situation in which one should take into account many variables present in the current situation. If we take into account human capabilities, using the cognitive architectures can allow for mechanical support of the activities in which humans are limited [12].

Cognitive architectures are expected to be able to solve not only one problem using the designated method, but also to be able to analyze the problem from a broader perspective, using a wider range of solutions. In addition, they will be able to implement new solutions in a dynamically changing environment with the help of acquired experience and knowledge. The acquisition and use of acquired knowledge is special here. Thus, cognitive architectures should consist of the following [56]:

- Memory that will store knowledge.
- Adequate representation of the knowledge.
- Processing units whose task will be to extract, combine, and store available information; they are often called operators that perform functions that “cause something to happen”; from the point of view of architecture, it does not matter if these functions describe the physical world, the imaginary world (e.g., human mind/imagination), or any virtual world generated for the chosen architecture.

Common important features of cognitive architectures are: how their learning mechanisms are designed and how their memory is organized. A large variety can be found in how the role of memory is perceived in the various approaches to building architectures [12].

Learning mainly consists in the fact that the acquired knowledge will be remembered and the possibilities and situations in which it can be used are assigned. Effective design and use of this process is a significant step in building higher cognitive systems. The way in which memory is organized is related to how knowledge is represented [12].

If we focus on these two important features, we can divide architectures into three types [57]:

- Symbolic
- Emergent
- Hybrid

Figure 49.1 presents the examples of such architectures. Each type of cognitive architecture can be represented as a series of steps required by architecture to perform the desired action (cognitive cycle). Although the term cognitive cycle is used to describe most architectures, they differ in the type of activities performed at each stage. The way they are grouped depends on the type of information presented and processed, and not on the various possibilities, properties, and criteria for assessing a given architecture [57].

49.4.2 Symbolic Architecture

Symbolic architectures represent concepts using symbols that can be manipulated using a predefined set of

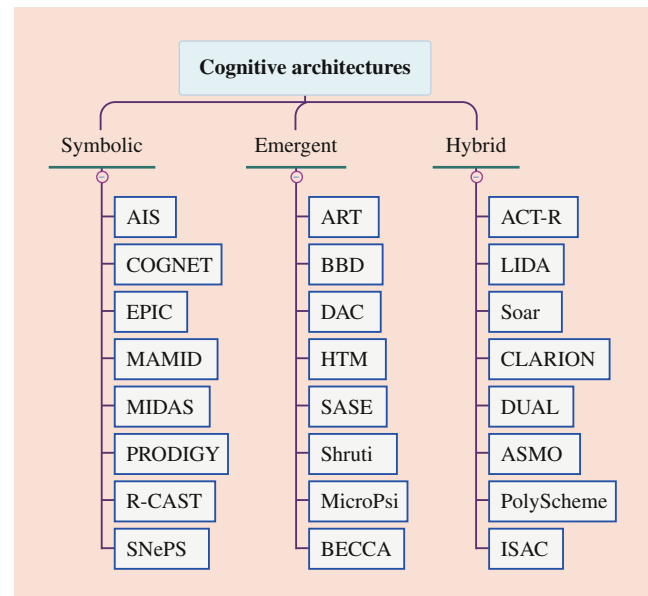


Fig. 49.1 Examples of cognitive architectures

instructions [57]. In most cases, this type of architecture uses centralized control of the flow of information from sensors to effectors. The executive functions related to the working memory are used here. In addition, knowledge stored in the semantic memory is used. This memory can be organized by means of production rules [12]. These rules are a very natural and intuitive representation of knowledge, therefore these are used by many practical solutions. Symbolic architectures stand out in planning and reasoning and they are able to cope with flexibility and resilience that enable us to cope with the changing environment and perceptual processing [57]. They work based on expert knowledge, after perceiving and identifying a problem. They are able to generate specific actions in line with assumed objectives of an action using their own knowledge.

Another organization of memory is graph representations, i.e., a structure consisting of vertices and edges that connect vertices. Usually, directed graphs are used where the edges immediately precede the vertices, and the vertices are direct successors of the edges. By transforming data into graphs, you can simplify the network to a large extent and create a clearer tool for solving problems [58].

Symbolic systems represent concepts using symbols that can be manipulated using a predefined set of instructions. Such instructions can be implemented as if-then rules applied to symbols representing facts known around the world (e.g., ACT-R, Soar, and other architecture of production rules).

The process of learning can be divided into an analytical (deductive) one and an inductive one. Analytical learning works on the principle of creating new facts based on agent’s knowledge. Inductive learning relies on finding the right

hypothesis among the observed data, which would agree or translate with already existing knowledge [12].

49.4.3 Emergent Architectures

An emergent architecture is not yet such an effective tool in solving problems, nor does it represent complex forms of knowledge like the symbolic architecture. This architecture sooner solves problems based on the interpretation of the observed data that reaches the system and the problems of perception. Neurobiological inspirations of emergent architectures appear in the form of connectionist and neural models. The network is composed of signal processing elements that represent specific memory traces, so one configuration is responsible for specific arousal in neural networks. Transitions between configurations are determined by connections between network nodes. In the neural model, based on the stimuli of several neurons (which individually fulfill specific functions), the system may interpret this configuration as a memory trace, e.g., of a specific object. There are associations between configurations, so that activation can pass to subsequent configurations of neurons. The actions occur between elements of the network. Changes take place in the entire network and new properties are developed [12].

The organization of memory in emergent architectures is carried out in two ways. One way is the global, distributed memory. All parameters occurring during the program operation have a direct impact on the achieved result. The other is local memory. The nodes of the network are activated by stimulation, so that only a few affect the final result [12].

Learning, in turn, is divided into an associative one and a competitive one. Associative learning transforms input data into simplified representations of output data, which can be divided into already known classes. The data are associated as objects, properties are matched to the possessed knowledge. Learning can be done under supervision, i.e., the data provided is compared to the expected network responses or by learning with the critic where the system is accounted from the few steps it has been able to perform (the result can be positive or negative). Competitive learning follows the “winner takes everything” principle, where the victorious element is the model for the next series of signals. The other elements are slowed down to simplify the signal flow [13].

This architecture solves the problems of system adaptation and learning by building parallel models, analogously to neural networks, in which the flow of information is initiated by signals from input nodes. However, the transparency of the system is low because it is no longer a set of symbolic entities, but it is distributed throughout the entire network. Hence, a logical inference in the traditional sense becomes problematic and unintuitive, which creates a new challenge to overcome for modern architectures [57].

49.4.4 Hybrid Architecture

The hybrid architecture combines the advantages of previous architectures, using effective planning and reasoning similar to the symbolic architecture, and good transformation of a large number of diverse data allowing the implementation of higher cognitive functions, similar to emergent architectures [12].

Memory organizations can be divided into a sprawled locale that includes a combination of local and distributed memory modules. The second division is symbolic-connectionist memory, which can be a combination of connectionist, dispersed, or local modules and symbolic modules [57].

Learning is divided into top-down or bottom-up domination. In top-down learning, the subsymbolic level determined by the lower level is taught by observing the dependencies from the received information from the higher (symbolic) level, where logical rules or other mechanisms of inference are operated. Teaching bottom-up mechanism works the opposite way. From the subsymbolic level, knowledge is transferred to the symbolic level in the form of formulated concepts [13].

Based of different types of architectures the cognitive agents are built.

49.4.5 Characteristic of Cognitive Agents

Cognitive agents internally regulate their goals based on their beliefs (like their decisions and plans). Goals and beliefs are a cognitive representation that can be internally generated, manipulated, and subject to reasoning [58, 59].

Most cognitive models try to model a certain type of cognitive process, e.g., perception, attention, memory, emotions, decision-making, choice of action, etc. or some narrow range within one of them. A much less frequent model at the level of systems (cognitive architecture) tries the full range of activities from incoming stimuli to outbound activities, along with the full range of cognitive processes between them [14].

Beliefs (knowledge possessed), expectations, goals, theories (coherent and explained set of beliefs, e.g., rules of production), plans, and intentions are representations used by cognitive agents. These representations affect the actions of themselves and other agents. A cognitive agent is defined in various ways in the literature, but the most common definition is that this type of agent is a computer program that [57]:

- Is capable of taking action in the environment in which it is located
- Can communicate directly with other agents

Table 49.2 Comparison of the cognitive and reactive agent

Cognitive agent	Reactive agent
Inference based on logic, ontologies, and knowledge	Inference based on the state of sensory inputs
Higher maintenance costs	Low maintenance costs
Self-sufficiency	Lack of self-sufficiency
Actions initiated by environmental stimuli	Actions depending on specific situations
Cognitive skills	Lack of cognitive skills
Learning	Functioning based on programmed methods, no learning opportunities

Source: own

- Is directed by a set of habits, tendencies that are specific goals, or an optimized function of benefits
- Has its own resources
- Is able to receive stimuli from its environment, but to a limited extent
- May have partial knowledge of this environment
- Has skills and can offer services
- Can reproduce, clone
- Displays behavior which urges it to strive to achieve goals taking into account available knowledge, resources and skills, and relying on the ability to receive stimuli from the perception and communication
- Can learn about the environment and learn by gaining experience

Therefore, it can be said that a cognitive agent is an intelligent program that not only requests based on the data received, takes specific actions to achieve the set goal (it can be, for example, supporting decision-making), but also, unlike a reactive agent, teaches while gaining experience. Table 49.2 presents the differences between cognitive agents and reactive agents.

It should be noted that the cognitive agent is self-sufficient, i.e., it can make its own decisions without human intervention (such agents are used, for example, in devices operating in space).

Higher costs of maintaining a cognitive agent are primarily associated with a greater demand for IT resources (hardware, software, and data resources).

In this chapter, attention will be paid to several types of cognitive agent architectures that we analyzed in our research:

- BOID
- Cougaar
- Soar
- SNePS
- LIDA

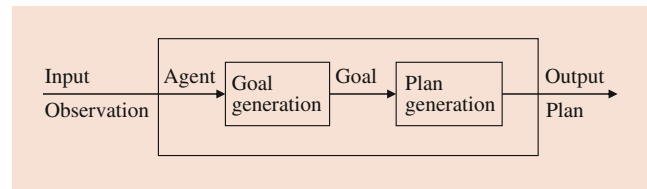


Fig. 49.2 Agent diagram in BOID architecture. (Source: own)

The first is the BOID architecture (beliefs, obligations, intentions, and desires).

Beliefs reflect the agent’s knowledge of the world around it, desires are defined as the states of the world – the potential goals of the agent, and its responsibilities are determined by tasks resulting from functioning in the group, while intentions reflect the agent’s real goals. The agent observes the environment and reacts appropriately to its changes using detectors and working mechanisms. Each object in the environment, called a component, is a process of input and output nature and operates based on rule systems that contain executable rule sets. The outputs represent the so-called “mental attitude” depending on the state of the inputs [59, 60].

Two modules are distinguished in this agent architecture (Fig. 49.2):

1. A goal-generation module that generates goals based on beliefs, desires, intentions, and responsibilities.
2. A plan-generation module that generates action sequences based on the generated goals.

In the case of agents supporting decision-making, goals and intentions are to help the decision-maker make decisions or to make the decisions independently (e.g., granting a loan and purchase shares), plans specify the actions to be taken to achieve the goal (e.g., read data and process it), beliefs express the decisions taken in the application process (for example, grant a loan and buy shares of given companies), while the obligations may specify that, for example, if a loan was granted to person 1, then the loan should also be granted to person 2 if he/she meets the same conditions as person 1 [61]. The BOID architecture is a symbolic architecture and can be used for financial decision supporting, but the frameworks of these architectures are too general and many of mechanisms for input data processing must be implemented by users.

Another cognitive architecture is Cougaar (cognitive agents architecture), which enables the creation of multi-agent systems using the Java programming language, developed by the ALPINE consortium for the DARPA (Defense Advanced Research Projects Agency). It consists of numerous layers of interactive applications – systems within systems [62]. This enables agents to be connected

in different communities [63]. For example, if we consider an enterprise, it may constitute a community, where the departments of the enterprise will be smaller communities located lower in the system hierarchy, while the counterparts of individual employees will be specific agent programs. In architecture, it is assumed that agents perform operations on so-called components (in other words, objects; for example, securities and currencies). The Cougaar architecture can therefore be used in the construction of multi-agent financial decision support systems, however it is a “black-box” and modification of software code (for example, to adapt to the field of the problem) is very difficult [64].

One of the more popular architectures is Soar (state, operator, and result). It is a general cognitive architecture for creating systems that exhibit intelligent behavior. Scientists, both in the fields of artificial intelligence and cognitive science, use Soar to perform a variety of tasks. It has been in use since 1983, evolving through a number of different versions. The creators of Soar have an ambitious plan to create a general artificial intelligence agent that will cope on many levels [65].

Plans for the functioning of Soar architecture provide for:

- Working on the full range of significant problems, open problems
- Using memory modules such as procedural, semantic, episodic, and iconic memory
- Use a full range of problem-solving methods
- Appropriate operation in an external environment
- Ability to solve tasks and implement them in the environment, combined with continuous learning about this process

Total rationality is a direction for the development of artificial intelligence. This means that the system, with the extensive knowledge it would have, would be able to take on any task it encounters. However, this goal is very difficult to achieve, so all attempts are now limited to an approximate version of rationality [65].

In Soar, decisions are made through current analysis and interpretation of sensory data provided to the agent. Working memory contains information needed to solve problems, often in the form of production rules or graphs, and all this is based on knowledge accumulated in long-term memory, e.g., semantic memory. Soar is characterized by functionality and efficiency. Functionality is that Soar has primitive cognitive skills that are needed to carry out tasks in accordance with the pattern of human reasoning. Among other things, it is able to make decisions based on reactions, understands the environment in which it works, plans, uses several forms of learning, and is aware of the situation. On the other hand, efficiency is based on the fact that it is able to take advantage

of the system’s advantages, such as efficient calculation of learned algorithms, has quick and easy access to saved experiences, is able to make decisions faster, and acquires and stores new knowledge.

In Soar there is a spatial visual system (SVS), independent from the system, which supports the interaction between connections needed for perception and motor control, and the symbolic representations of these relationships. In addition to internal support, SVS also supports the system externally. Thanks to this, the agent can use so-called mental images. They allow continuous interpretation of a dynamically changing environment. This is useful, e.g., in the use of car controls, with modern algorithms that interpret the planning of its path. Soar belongs to hybrid architecture; it uses both symbolic and nonsymbolic structure [66]. The working memory in the symbolic structure contains current and recent sensory data, the current goal, and interpretations of the situation in the context of implementation of this goal. Working memory acts as a buffer for long-term memory and motor function. The semantic memory contains information that illustrates the agent’s model of the external world, but it is downloaded to the working memory when it is needed. Newly created information in the learning process is delivered to the semantic memory, although one can implement the rules at the beginning of the agent’s creation. In the episodic memory, on the other hand, there are snapshots of the agent’s experience. This memory accumulates experience and allows one to use it to guide future behaviors and draw conclusions. Among numerous cognitive agent programs, this type of memory is not frequently employed, but it is extremely useful because it leads to maintaining greater consistency between experiences [66].

The procedural memory contains production rules. Specific situations are coded there, including descriptions as to how the system should deal with them. Processing units are operators which arise in working memory and propose solutions based on the current situation. They have their preferences, which are analyzed using established decision-making procedures. The agent determines whether the operator is suitable to participate in the cognitive cycle, and if not, it causes stagnation. This contributes to the fact that the operator enters the substate in which there may be more thought-out reasoning, task decomposition, changes in planning, and methods of finding solutions. The selected operator modifies working memory, e.g., by downloading from semantic or episodic memory, or by issuing external motor commands [65].

After selecting the operator, rules sensitive to his/her choice perform their actions by modifying the working memory. The action can be an internal step of reasoning, a query to SVS, downloading from episodic or semantic memory, or an external motor command. The operators in Soar conceptually correspond to the STRIPS operators; however, in

Soar the constituent parts of the operators (preconditions and actions) are broken down into individual rules to ensure the preconditions and conditional disjunctive actions. In addition, Soar has rules which are dependent on the tasks to assess the operator, ensuring contextual and precise control of operator selection and application [65]. The Soar can be used in multi-agent financial decision support systems, however it is characterized by high demand for hardware resources.

Another example of cognitive agent programs is SNePS – the Semantic Network Processing System. It is a system based on logical, framework, and network representations of knowledge, inference, and action. Semantic knowledge works on the principle of ordered direct graphs whose nodes represent entities and arches represent the relationship between two nodes [66]. The unit represented by node n reacts with the R arch with the unit represented by node m . In SNePS, a set of nodes in the program network and a set of logical conditions operate at the same time. When a problem arises that needs to be resolved, the entire SNePS operation is implemented at the node level, not at the arches. The relationships represented by arches can be understood as part of the structure of nodes that illustrate where certain activities originate. Each time information is added to the network, new connections are created that are implemented with the principle of operation of nodes and arches. New information is assigned to the appropriate relationship. Each autonomous representation is expressed as a node or condition. Each unit in the network is represented by a unique node. The program user can add such representations according to their needs [66].

There are four types of nodes [66]:

- Basic – they do not originate from arches, i.e., they do not start further action; it is assumed that the basic nodes represent, e.g., a unit: objects, classes, properties of entities; they can be ranked according to an equality or equivalence relationship; and it is also possible to specify the rules for using these nodes.
- Variables – additionally, they do not start an activity, but they represent any units or suggestions in a similar way as logical variables do.
- Molecular – they possess arches which extend from them and they can represent various solutions, including rules.
- Reference – they also possess arches coming out from them; they represent any sentences or units of any structure; and they are similar to open tasks in the predicate logic.

An important function of the SNePS program is the introduction of context functions. It is a structure consisting of three components [67]:

- A set of hypotheses, i.e., a set of nodes from which the context is composed; the hypotheses are defined in such a way that two contexts will not have the same set.
- A set of names, i.e., symbols that represent the context.
- A controller checking if a set of hypotheses for the structure of this context may be conflicting, in which case it returns TRUE.

The agent having the entire structure in the semantic network is supported by the SNIP: the SNePS inference package, which can be automatically applied. Whereas SNeRE: the SNePS rational engine is a package responsible for planning and taking action. Also available is the SNaLPS package: the SNePS natural language processing system, which efficiently analyses natural language [66].

SNePS agents are used for linguistic analysis, as they possess the ability to expand vocabulary and create text responses [12]. They also find their application as data managers. SNePS works in specific databases, where there are numerous connections between data tables and when the user can search and organize data in many ways. This is a more natural form of approach to working with data, because the system is able to better understand the relationships between them. Elements of such databases are represented by primary nodes, rows are represented by molecular nodes, and arches represent columns [66]. SNePS can be used in multi-agent decision support system, however its documentation is very weak.

In our research we use solution proposed by S. Franklin – the Learning Intelligent Distribution Agent (LIDA). Its architecture is shown in Fig. 49.3. A cognitive agent consists of the following modules [13]:

- Workspace
- Global workspace
- Sensory memory
- Perceptual memory
- Episodic memory
- Declarative memory
- Sensory and motor memory
- Selection of actions

LIDA is an agent that works at the system level. It receives many simultaneous incoming stimuli, and then internally processes them, which is completed by choosing and performing the appropriate action. The agent creates hypotheses, acts as a cognitive help in the thinking and understanding of individual cognitive activities and various executive processes. As a model, it tries to explain and predict the phenomena occurring in the environment. The basic function of the agent is to constantly answer the question: “What should I do next?” [13]. The agent’s operation is conditioned by the stimuli it receives from the external or internal environment.

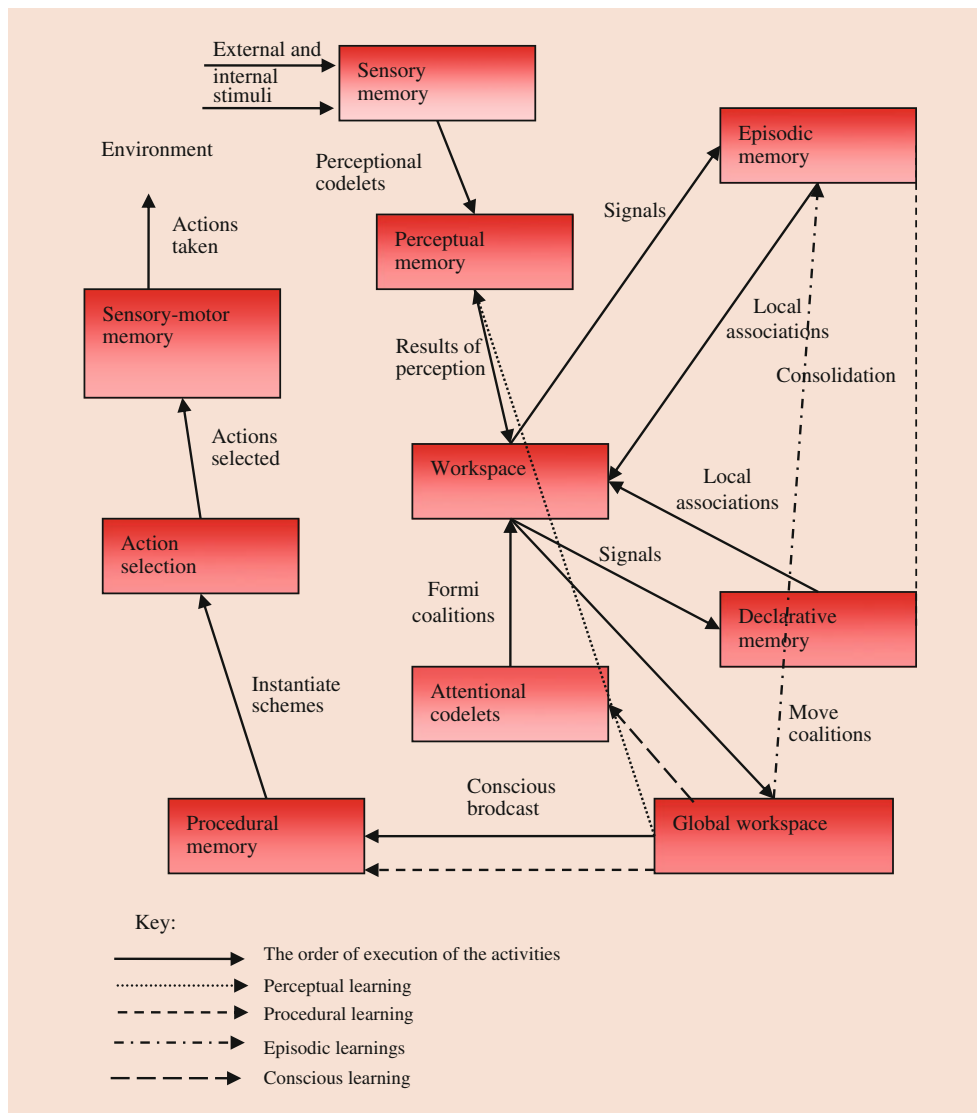


Fig. 49.3 The LIDA architecture

The key is that its reaction results from the cognitive cycle, which is the core of the agent's activities. The cycle consists of the higher-order cognitive processes, which include reflection, reasoning, problem-solving, planning, imagining, etc. It is divided into three phases [13]:

- The phase of perception and understanding – after receiving input data from, for example, motor sensors or agent's memories, it tries to understand the actual situation.
- The phase of consciousness – in this phase the content of this understanding is filtered, because the agent is interested in keeping only relevant information that it will be able to send to global workspace.
- The phase of action and learning – in this phase the system selects and then performs selected actions and learns in many memory mechanisms.

The LIDA cognitive cycle, presented in Fig. 49.1, begins with sensory stimuli, both external and internal, reaching the sensory memory, where early feature detectors are involved. Received data goes to perceptual memory. In the perceptual memory an association model is created, which aims to create perception and make it available to the workspace, which is constantly updated by sending signals to the perceptual, declarative, and episodic memory and receives local associations. The entire update happens in workspace by processing units called codelets. These small pieces of code perform a specific task when the expected conditions are met [13].

In the phase of consciousness, the codelets browse the content contained in the workspace and decide which of them should be introduced as a signal of consciousness delivered to global workspace. There, on the other hand, these data form coalitions, which, through competition, become

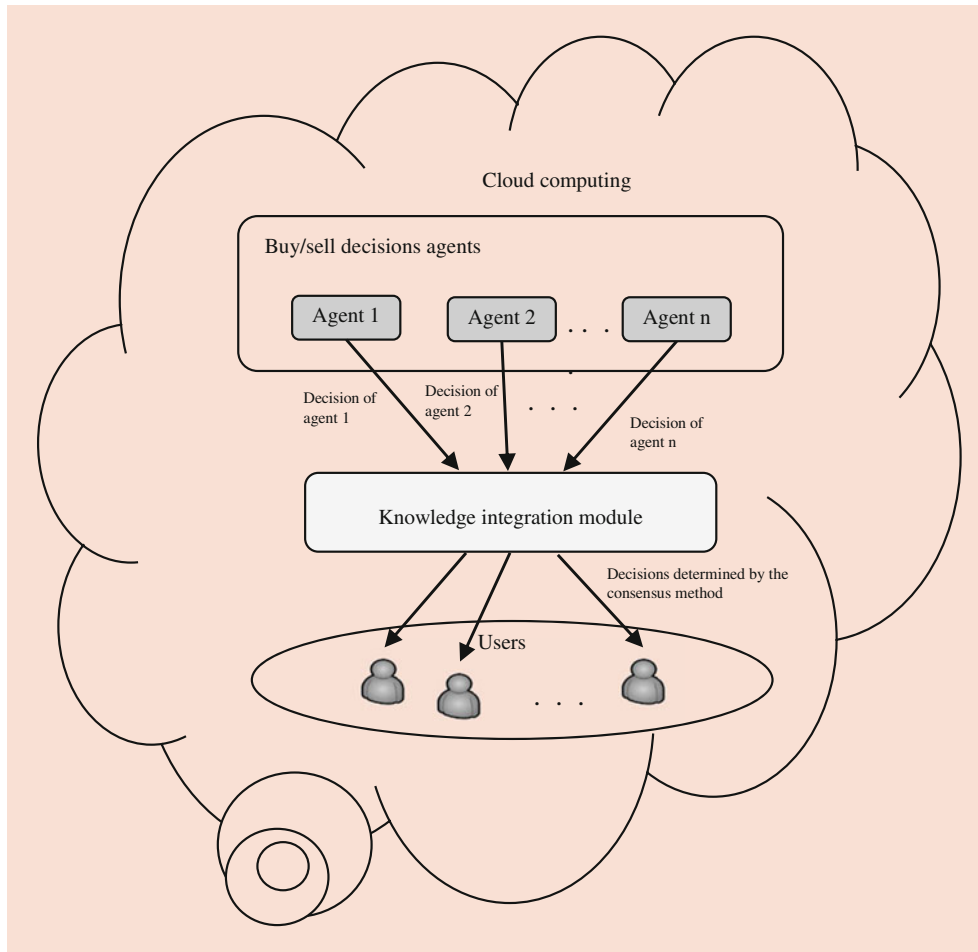


Fig. 49.4 Functional architecture of MAFDSS

selected as the right consciousness. The winning, most significant coalition has its own content transmitted globally. It becomes aware in the functional sense, completing the phase of attention of the cognitive cycle [68].

In the last, third phase of the cognitive cycle, almost every module can get the appropriate content of consciousness that matches its data structure for learning. For example, the procedural memory, which is responsible for “what to do when . . .” knowledge, uses information that in the future will facilitate an immediate response to over-stimuli. In the final stages of the cycle, the action selection module selects one action from the schemas provided to it, then it is sent to the sensory-motor memory to create or select an appropriate action plan that can then be performed. This action completes the cognitive cycle LIDA [69].

49.5 The Architecture and Functionality of MAFDSS

The purpose of MAFDSS is to support investing in financial markets by subcontracting the investor to buy/sell

decisions. The system consists of the following elements (Fig. 49.4):

- Agents making buy/sell decisions that process structured and unstructured knowledge.
- Knowledge integration module – operates using the consensus method; in this module, based on decisions generated by agents, a final decision is presented to the user. The issues related to this module has been presented by Hernes et al. [69, 70] and they are not a subject of this chapter.
- Users – people investing in the financial market.

The buy/sell decisions agents are divided into following groups:

- Technical analysis agents (they have been described, for example, by Korczak et al. [71])
- Fundamental analysis agents [72]
- Behavioral analysis agents [71]
- Deep learning agents [73]
- Text analysis agents

In this chapter, we focus on text analysis agents. The main aim of these agents is to offer advice on buy/sell decision based on expert or investors' opinions stored in cyberspace (e.g., portals, forums, and Tweeter).

The environment of the text analysis agent functioning is a set of text documents containing such opinions. The analysis of the opinion is carried out as follows:

1. On the basis of a training set (a set of opinions concerning a given market, e.g., Forex), a semantic network is created in the perceptual memory containing concepts (related to the product) and connections (associations) between them. The perceptual memory also stores synonyms and various types of words. In the perceptual memory of the LIDA agent, concepts are represented by nodes, while associations by links.
2. Sensory memory (containing strings) is passed in turn to individual opinions.
3. Opinion analysis is made by codelets (implemented in the form of Java programming language classes). They analyze the text according to the criteria defined by the configuration parameters (stored in the xml file). The values of parameters can be indicated by the user and used in a codelet software code. An example of a codelet configuration defining the sentiment of the opinion is presented in Fig. 49.5.

The task name parameter (in the LIDA codelet architecture is configured as a task, the task may also be to refresh the GUI content and it is not implemented by the codelet) means the codelet name. The task type parameter indicates the name of the Java class in which the codelet's program code is located, the parameter-named object specifies which words (or exceptions) are searched by the codelet in the sensory memory. The no-object parameter specifies which words (or expressions) cannot be in the text (for example, the opinion has buy

```
<task name="buy">
<tasktype> CodeletObjectDetector </tasktype>
<param name="object" type="string">sold out</param>
<param name="object" type="string">market</param>
<param name="noobject" type="string">bought
out</param>
<param name="distance" type="int">2</param>
<param name="node" type="string">buy</param>
</task>
```

Fig. 49.5 The example of the configuration of codelet

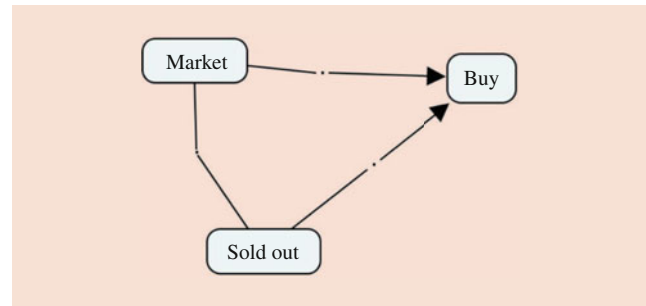


Fig. 49.6 The example of results of opinion analysis

- sentiment when the word “sold out” appears in the text, but the word “bought out” does not appear). The distance parameter specifies the maximum distance between the searched expression or expressions, the node parameter specifies which node has to be placed in the workspace in the case of finding the wanted items (or expression) [74–76].
4. The results of the analysis, in the form of a semantic network, are transferred to the working memory (the current situational model is created). Figure 49.6 shows an example of the results of the analysis of the following Product Opinion 1: “The market is sold out.” Nodes are marked with a large circle symbol, and links with the arrow symbol. The dots indicate levels of link activation (links can be determined with a certain level of probability).
 5. In the next step, the situational model is passed to the global workspace and the following action schemes are automatically selected from the procedural memory: “writing the results of the analysis of the opinion to the database” (noSQL database – analysis results – semantic network – are saved in the XML format) and “loading another opinion into sensory memory.” It is also possible to select the action “statistical analysis,” as a result of which the agent also indicates the features of the product that are most desirable by customers.

The next part of the chapter presents the results of performance evaluation of text analysis agent on the Forex market.

49.6 Research Experiment

The analysis of the efficiency of the developed method was carried out on data from the H4 range of quotations from the Forex. In order to make this analysis, a test was carried out in which the following assumptions were made:

1. The GBP/PLN pair quotations were used from three randomly selected periods:

- 17-04-2018 0:00 to 23-04-2018 23:59
 - 15-05-2018 0:00 to 21-05-2018 23:59
 - 28-05-2018 0:00 to 01-06-2018 23:59
2. The verification was based on the opinions of experts obtained from financial portals. These opinions were analyzed by the cognitive agent and on the basis of the analysis results, the agent determined which decisions should be taken in the periods considered (buy value 1, sell value -1 , leave unchanged value 0).
 3. It has been assumed that the unit of measure of effectiveness (relative measures) is pip (price change by one "point" on the Forex market is referred to as pips).
 4. Transaction costs are not taken into account.
 5. Money management – it was assumed that in each transaction, the investor engages 100% of his capital. The capital management strategy can be determined by the user. The efficiency analysis was carried out using the following measures (indicators):
 - Rate of return (ratio x_1)
 - Number of transaction
 - Gross profit (ratio x_2)
 - Gross loss (ratio x_3)
 - Total profit (ratio x_4)
 - Number of profitable transaction (ratio x_5)
 - Sharpe ratio (ratio x_6)

$$S = \frac{E(r) - E(f)}{|O(r)|} \times 100\% \quad (49.1)$$

where

$E(r)$ – the arithmetic average of the rate of return

$E(f)$ – the arithmetic average of the risk-free rate of return

$O(r)$ – standard deviation of return rates

- Average coefficient of variation (ratio x_7) is the ratio of the average deviation to the arithmetic mean multiplied by 100%:

$$V = \frac{s}{|E(r)|} \times 100\% \quad (49.2)$$

where

V – average coefficient of variation

s – deviation of the average rate of return

$E(r)$ – the arithmetic mean of the rate of return

6. The following performance evaluation function will be used [42]:

$$y = \left(a_1x_1 + a_2x_2 + a_3(1 - x_3) + a_4x_4 + a_5x_5 + a_6x_6 + a_7(1 - x_7) \right) \quad (49.3)$$

where x_i denotes normalized values of the indicators.

The results obtained on the basis of text analysis agent have been compared with the results of the fundamental analysis agent (based on fundamental analysis method – described in detail by Korczak et al. [72]), technical analysis agent (based on different technical analysis indicators – described in detail by Korczak et al. [71]), and buy-and-hold strategy (the investor makes a decision to buy at the beginning of the period, and the decision to sell at the end of the period). Table 49.3 presents the results of research experiment.

Analyzing the results, it can be stated that the values of particular ratios are characterized by a high dispersion. There is no agent which achieved the best values of particular ratios in all the periods. Taking into consideration a *rate of return*, the text analysis agent achieved the highest value in period 1 and 3. In period 2, the technical analysis agent achieved the highest value of this ratio. The B & H is characterized by the lowest values of *rate of return* in all periods. The technical analysis agent generates the highest number of transactions in each period. On the basis of values of *gross profit* and *gross loss* we can conclude that even if the overall *rate of return* in the given period is profitable, there are transactions generating losses. Taking into consideration the value of the *number of profitable transactions* it can be stated that this value was 50% or higher only in the case of the text analysis agent in all periods. The technical analysis agent generated the highest number of unprofitable transactions in relation to other agents. A large number of unprofitable transactions can lead to losses in investment capital even if the overall *rate of return* is profitable. Risk-based measures are very important – Sharpe ratio (the risk is lower if the ratio is higher) and average coefficient of volatility (the risk is lower if the ratio is lower). The text analysis agent and the fundamental analysis agent achieve the best values of these ratios. The transactions generated by these agents are less risky than transactions generated by the technical analysis agent.

The analysis of particular ratios by a user is a very time-consuming process. Therefore, the evaluation function is very helpful in this purpose. Taking into consideration this function, it can be stated that in period 1 and 3 the text analysis agent achieved the highest value, the fundamental agent was ranked higher in the second period. The B&H benchmark was ranked lowest in all the periods.

49.7 Conclusions

The application of a cognitive architecture in multi-agent systems supporting taking financial decisions allows performing it on the basis of natural language processing. Decisions generated by cognitive agent on the basis of opinions of

Table 49.3 Results of research experiment

Ratio	Text analysis agent			Fundamental analysis agent			Technical analysis agent			B & H		
	Period 1	Period 2	Period 3	Period 1	Period 2	Period 3	Period 1	Period 2	Period 3	Period 1	Period 2	Period 3
Rate of return [Pips]	-274	9345	2487	-675	8237	2389	-578	9373	2285	-1016	7246	1996
Number of transactions	4	7	5	3	6	6	11	16	9	1	1	1
Gross profit [Pips]	378	1273	378	429	2419	437	214	942	194	0	7246	1996
Gross loss [Pips]	-421	-438	-261	494	-685	-294	-396	-582	-146	-1016	0	0
Number of profitable transactions	2	5	4	1	5	4	4	10	5	0	1	1
Sharpe ratio	0.78	0.82	0.63	0.74	0.80	0.72	0.36	0.63	0.59	0.27	0.46	0.17
Average coefficient of volatility	0.28	0.36	0.32	0.31	0.28	0.42	0.47	0.43	0.58	-	-	-
Average rate of return per transaction	-68.5	1335.0	497.4	-225.0	1372.8	398.2	52.55	585.8	253.9	-1016	7246	1996
Value of evaluation function (y)	0.67	0.52	0.59	0.51	0.58	0.54	0.38	0.41	0.46	0.14	0.21	0.17

experts allow for achieving more satisfactory benefits in relation to, for example, the fundamental analysis agent or the technical analysis agent. It can result from the fact that experts build their opinions on the basis of different methods (including fundamental, behavioral, and technical analysis). Expert's opinions also consist of the market sentiment. It is highly probable that the "wisdom of the crowd" phenomena occurs also in this case.

To sum up, the cognitive architecture allows to increase the usability of the multi-agent system and consequently to improve the process of taking investment decisions. The ability to make automatic decisions is also of high importance here. For example, the cognitive agent can make real transactions (open/close short/long position) on the Forex market.

The further research works can be related, for example, to implementing deep text analysis method into the cognitive architecture and to developing an agent running on the basis of summarization of opinions of experts and users.

Acknowledgments The project is financed by the Ministry of Science and Higher Education in Poland under the program "Regional Initiative of Excellence" 2019–2022 project number 015/RID/2018/19 total funding amount 10 721 040,00 PLN.

References

- Jajuga, K., Orłowski, L.T., Staehr, K.: Contemporary Trends and Challenges in Finance. Springer International Publishing, Cham (2017)
- Hernes, M., Sobieska-Karpińska, J.: Application of the consensus method in a multiagent financial decision support system. *Inf. Syst. E-Bus. Manag.* **14**(1), 167–185 (2016)
- Mondal, A., Singh, A.: Emerging technologies and opportunities for innovation in financial data analytics: a perspective. In: Mondal, A., Gupta, H., Srivastava, J., Reddy, P., Somayajulu, D. (eds.) *Big Data Analytics. BDA 2018 Lecture Notes in Computer Science*, vol. 11297. Springer, Cham (2018)
- Dapp, T., Slomka, L.: Fintech reloaded – traditional banks as digital ecosystems (Publication of the German original), pp. 261–274 (2015)
- Kliman, R., Arinze, B.: Cognitive computing: impacts on financial advice in wealth management. In: *Aligning Business Strategies and Analytics*, pp. 11–23. Springer, Cham (2019)
- Schumaker, R.P., Zhang, Y., Huang, C.N., Chen, H.: Evaluating sentiment in financial news articles. *Decis. Support. Syst.* **53**(3), 458–464 (2012)
- Ahmad, K., Cheng, D., Almas, Y.: Multi-lingual sentiment analysis of financial news streams. In: *1st International Workshop on Grid Technology for Financial Modeling and Simulation*, 26, (SISSA Medialab) (2007)
- Jung, D., Dorner, V., Weinhardt, C., Puzmaz, H.: Designing a robo-advisor for risk-averse, low-budget consumers. *Electron. Mark.* **28**(3) (2018). <https://doi.org/10.1007/s12525-017-0279-9>
- Alt, R., Beck, R., Smits, M.T.: FinTech and the transformation of the financial industry. *Electron. Mark.* **28**(3), 235–243 (2018). <https://doi.org/10.1007/s12525-018-0310-9>
- Serrano, W.: The random neural network with a genetic algorithm and deep learning clusters in fintech: smart investment. In: Iliadis, L., Maglogiannis, I., Plagianakos, V. (eds.) *Artificial Intelligence Applications and Innovations. AIAI 2018. IFIP Advances in Information and Communication Technology*, vol. 519. Springer, Cham (2018)
- Sohangir, S., Wang, D., Pomeranets, A., Khoshgoftaar, T.M.: Big Data: deep learning for financial sentiment analysis. *J. Big Data.* **5**(3) (2018). <https://doi.org/10.1186/s40537-017-0111-6>
- Duch, W., Oentaryo, R.J., Pasquier, M.: Cognitive architectures: where do we go from here? In: Wang, P., Goertzel, P., Franklin, S. (eds.) *Frontiers in Artificial Intelligence and Applications*, vol. 171, pp. 122–136. IOS Press (2008)
- Sowa, J.F.: The cognitive cycle. <http://www.jfsowa.com/pubs/cogcycle.pdf>

14. Franklin, S., Patterson, F.G.: The LIDA architecture: adding new modes of learning to an intelligent, autonomous, software agent. In: *Proceedings of the International Conference on Integrated Design and Process Technology*. Society for Design and Process Science, San Diego (2006)
15. Luo, Y., Liu, K., Davis, D.N.: A multi-agent decision support system for stock trading. *IEEE Netw.* **16**(1), 20–27 (2002)
16. Hafezi, R., Shahrabi, J., Hadavandi, E.: A bat-neural network multi-agent system (BNNMAS) for stock price prediction: case study of DAX stock price. *Appl. Soft Comput.* **29**, 196–210 (2015)
17. Kim, S.: Hybrid forecasting system based on case-based reasoning and analytic hierarchy process for cost estimation. *J. Civ. Eng. Manag.* **19**(1), 86–96 (2013)
18. Gottschlich, J., Hinz, O.: A decision support system for stock investment recommendations using collective wisdom. *Decis. Support. Syst.* **59**, 52–62 (2014)
19. Xia, J.: Multi-agent investment in incomplete markets. *Finance Stochast.* **8**(2), 241–259 (2004)
20. Roosen, P., Uhlenbruck, S., Lucas, K.: Pareto optimization of a combined cycle power system as a decision support tool for trading off investment vs. operating costs. *Int. J. Therm. Sci.* **42**(6), 553–560 (2003)
21. Barbosa, R.P., Belo, O.: Multi-agent forex trading system. In: Håkansson, A., Hartung, R., Nguyen, N.T. (eds.) *Agent and Multi-agent Technology for Internet and Enterprise Systems*, Studies in Computational Intelligence, vol. 289. Springer, Berlin, Heidelberg (2010)
22. Said, Y.B., Kanzari, D., Bezzine, M.: A behavioral and rational investor modeling to explain subprime crisis: multi agent systems simulation in artificial financial markets. In: Masri, H., Pérez-Gladish, B., Zopounidis, C. (eds.) *Financial Decision Aid Using Multiple Criteria*. Multiple Criteria Decision Making. Springer, Cham (2018)
23. Chen, S., Tien, J., Spotton Visano, B.: A hybrid multi-agent model for financial markets. In: Nguyen, N.T., Borzowski, L., Grzech, A., Ali, M. (eds.) *New Frontiers in Applied Artificial Intelligence*. IEA/AIE 2008 Lecture Notes in Computer Science, vol. 5027. Springer, Berlin, Heidelberg (2008)
24. O'Hare, N., Davy, M., Bermingham, A., Ferguson, P., Sheridan, P., Gurrin, C., Smeaton, A.F.: Topic-dependent sentiment analysis of financial blogs. In: *Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion*, pp. 9–16. ACM (2009)
25. Wang, Y.J.: Stock market forecasting with financial micro-blog based on sentiment and time series analysis. *J. Shanghai Jiaotong Univ. Sci.* **22**(2), 173–179 (2017). <https://doi.org/10.1007/s12204-017-1818-4>
26. Li, L., Goh, T.T., Jin, D.: How textual quality of online reviews affect classification performance: a case of deep learning sentiment analysis. *Neural Comput. & Applic., 1–29* (2018). <https://doi.org/10.1007/s00521-018-3865-7>
27. Chan, S.W., Chong, M.W.: Sentiment analysis in financial texts. *Decis. Support. Syst.* **94**, 53–64 (2017)
28. Westerhoff, F.H.: Multiasset market dynamics. *Macroecon. Dyn.* **8**, 1171–1180 (2011)
29. Korczak, J., Hernes, M., Bac, M.: Performance evaluation of decision-making agents' in the multi-agent system. In: Ganzha, M., Maciaszek, L., Paprzycki, M. (eds.) *Proceedings of Federated Conference Computer Science and Information Systems (FedCSIS)*. Warsaw (2014)
30. Olds, J.L.: Cognitive technology. In: Bainbridge, W., Roco, M. (eds.) *Handbook of Science and Technology Convergence*. Springer, Cham (2016)
31. Baldoni, M., Baroglio, C., Patti, V., Rena, P.: From tags to emotions: ontology-driven sentiment analysis in the social semantic web. *Intelligenza Artificiale.* **6**(1), 41–54 (2012)
32. Pham, L.V., Pham, S.B.: Information extraction for Vietnamese real estate advertisements. In: *Fourth international conference on knowledge and systems engineering (KSE)*, Danang (2012)
33. Phan, H.T., Nguyen, N.T., Tran, V.C., Hwang, D.: A method for detecting and analyzing the sentiment of tweets containing conditional sentences. In: *Intelligent information and database systems: 11th Asian conference, ACIIDS 2019, Yogyakarta, Indonesia, April 8–11, 2019: proceedings. Pt. 1 / Ngoc Thanh Nguyen [i in.]* (eds.). Springer, Cham, pp. 177–188 (2019)
34. Hassouna, M., Tarhini, A., Elyas, T., AbouTrab, M.S.: Customer churn in mobile markets a comparison of techniques. *arXiv preprint arXiv:1607.07792* 2016
35. Iqbal, N., Khattak, S.R., Khattak, M.A.: Does fundamental analysis predict stock returns? Evidence from non-financial companies listed on KSE. *Knowl. Horiz. Econ.* **5**(4), 182–190 (2013)
36. Chen, Y.J., Chen, Y.M.: A fundamental analysis-based method for stock market forecasting. In: *2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP)*, pp. 354–359. IEEE (2013)
37. Bayramoglu, M.F., Hamzacebi, C.: Stock selection based on fundamental analysis approach by grey relational analysis: a case of Turkey. *Int. J. Econ. Financ.* **8**(7), 178–184 (2016)
38. Wafi, A.S., Hassan, H., Mabrouk, A.: Fundamental analysis vs technical analysis in the Egyptian stock exchange: empirical study. *Int. J. Bus. Manage. Study.* **2**(2), 212–218 (2015)
39. Edwards, R.D., Magee, J., Bassetti, W.C.: *Technical Analysis of Stock Trends*. CRC Press (2018)
40. Kirkpatrick II, C.D., Dahlquist, J.A.: *Technical Analysis: the Complete Resource for Financial Market Technicians*. FT Press (2010)
41. Shynkevich, A.: Performance of technical analysis in growth and small cap segments of the US equity market. *J. Bank. Financ.* **36**(1), 193–208 (2012)
42. Wang, M., Chen, Y., Tian, L., Jiang, S., Tian, Z., Du, R.: Fluctuation behavior analysis of international crude oil and gasoline price based on complex network perspective. *Appl. Energy.* **175**, 109–127 (2016)
43. Chiang, T.C., Zheng, D.: An empirical analysis of herd behavior in global stock markets. *J. Bank. Financ.* **34**(8), 1911–1921 (2010)
44. Thaler, R.H., Ganser, L.J.: *Misbehaving: the Making of Behavioral Economics*. WW Norton, New York (2015)
45. Christoffersen, P.: *Elements of Financial Risk Management*. Academic Press (2012)
46. Waemustafa, W., Sukri, S.: Systematic and unsystematic risk determinants of liquidity risk between Islamic and conventional banks. *Int. J. Econ. Financ. Issues.* **6**(4), 1321–1327 (2016)
47. Kang, J., Pflueger, C.E.: Inflation risk in corporate bonds. *J. Financ.* **70**(1), 115–162 (2015)
48. Esposito, L., Nobili, A., Ropele, T.: The management of interest rate risk during the crisis: evidence from Italian banks. *J. Bank. Financ.* **59**, 486–504 (2015)
49. Lee, C.F., Wang, C.S., Xie, A.Y.: Exchange rate risk in the US Stock Market: a pooled panel data regression approach. Available at SSRN 3407704 (2018)
50. Hatchondo, J.C., Martinez, L., Sosa-Padilla, C.: Debt dilution and sovereign default risk. *J. Polit. Econ.* **124**(5), 1383–1422 (2016)
51. Semernina, Y.V., Yakunina, A.V., Nesterenko, E.A., Yakunin, S.V., Korobov, E.A.: Evaluation of reinvestment risk for bond portfolios. In: *CEUR Workshop Proceedings*, 184–192 (2018)
52. Morris, S., Shim, I., Shin, H.S.: Redemption risk and cash hoarding by asset managers. *J. Monet. Econ.* **89**, 71–87 (2017)
53. Shapiro, A.: Interchangeability principle and dynamic equations in risk averse stochastic programming. *Oper. Res. Lett.* **45**(4), 377–381 (2017)
54. Tonzler, L.: Cross-border interbank networks, banking risk and contagion. *J. Financ. Stab.* **18**, 19–32 (2015)

55. Albus, J.S., Barbera, A.J.: RCS: a cognitive architecture for intelligent multi-agent systems. *Annu. Rev. Control.* **29**(1), 87–99 (2005)
56. Laird, J.E.: Extending the SOAR cognitive architecture. In: Wang, P., Goertzel, B., Franklin, S. (eds.) *Frontiers in Artificial Intelligence and Applications*, vol. 171. IOS, Amsterdam (2008)
57. Kotseruba, I., Tsotsos, J.K.: 40 years of cognitive architectures: core cognitive abilities and practical applications. *Artificial Intelligence Review*, **53**(1), 17–94 (2020)
58. Samsonovich, A.V.: Toward a unified catalog of implemented cognitive architectures. *BICA.* **221**, 195–244 (2010)
59. Frantz, C.K., Pigozzi, G.: Modeling norm dynamics in multi-agent systems. *J. Appl. Log. ifCoLog J. Log. Their Appl.* **5**(2), 491–563 (2018)
60. Bădică, C., Budimac, Z., Burkhard, H.D., Ivanovic, M.: Software agents: languages, tools, platforms. *Comput. Sci. Inf. Syst.* **8**(2), 255–298 (2011)
61. Mahmoud, M.A., Ahmad, M.S., Mohd Yusoff, M.Z., Mustapha, A.: A review of norms and normative multiagent systems. *Sci. World J.*, 2014 (2014)
62. López, T.S., Brintrup, A., McFarlane, D., Dwyer, D.: Selecting a multi-agent system development tool for industrial applications: a case study of self-serving aircraft assets. In: 4th IEEE International Conference on Digital Ecosystems and Technologies, pp. 400–405. IEEE (2010)
63. Leon, F., Paprzycki, M., Ganzha M.: A review of agent platforms. *Multi-Paradigm Modelling for Cyber-Physical Systems (MPM4CPS), ICT COST Action IC1404*, pp. 1–15 (2015)
64. Shapiro, S.C., Rapaport, W.J.: A fully intensional propositional semantic network. *Philos. Comput.* **75** (2019)
65. Laird, J.E.: Extending the SOAR cognitive architecture. In: Wang, P., Goertzel, B., Franklin, S. (eds.) *Frontiers in Artificial Intelligence and Applications*, vol. 171. IOS Press, Amsterdam (2008)
66. Laird, J.E.: *The Soar Cognitive Architecture*. MIT Press, Cambridge (2012)
67. Konderak, P.: On a cognitive model of semiosis. *Stud. Log. Gramm. Rhetor.* **40**(1), 129–144 (2015)
68. Hernes, M.: Performance Evaluation of the Customer Relationship Management Agent's in a Cognitive Integrated Management Support System, *Transactions on Computational Collective Intelligence*, vol. XVIII, pp. 86–104. Springer, Berlin (2015)
69. Hernes, M., Bytniewski, A.: Integration of collective knowledge in financial decision support system. In: *Intelligent Information and Database Systems. 8th Asian Conference*, ed. by N.T. Nguyen, B. Trawiński, H. Fujita, T.-P. Hong, ACIIDS 2016, Da Nang, Vietnam, Proceedings, Part 1 (T. 9621, ss. 470–479). Springer, 14–16 Mar 2016
70. Hernes, M., Chojnacka-Komorowska, A., Kozierkiewicz, A., Pietranik, M.: Agents' knowledge conflicts' resolving in cognitive integrated management information system – case of budgeting module. In: Nguyen, N.T., Pimenidis, E., Khan, Z., Trawiński, B. (eds.) *Computational Collective Intelligence. ICCCI 2018 Lecture Notes in Computer Science*, vol. 11055. Springer, Cham (2018)
71. Korczak, J., Hernes, M., Bac, M.: Collective intelligence supporting trading decisions on FOREX market. In: Nguyen, N.T., Papadopoulos, G.A., Jędrzejowicz, P., Trawiński, B., Vossen, G., Nguyen, N.T., Papadopoulos, G.A., Jędrzejowicz, P., Trawiński, B., & Vossen G. (eds.) *Computational Collective Intelligence. 9th International Conference, ICCCI 2017, Nicosia, Cyprus, 27–29 Sept 2017, Proceedings, Part I*, Springer International Publishing (2017). https://doi.org/10.1007/978-3-319-67074-4_12
72. Korczak, J., Hernes, M., Bac, M.: Fundamental analysis in the multi-agent trading system. In: Ganzha, M., Maciaszek, L., Paprzycki, M., Ganzha, M., Maciaszek, L., Paprzycki, M. (eds.) *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems*, pp. 1169–1174. Polskie Towarzystwo Informatyczne; Institute of Electrical and Electronics Engineers, Gdańsk (2016). <https://doi.org/10.15439/2016F238>
73. Korczak, J., Hernes, M.: Deep learning for financial time series forecasting in A-trader system. In: Ganzha, M., Maciaszek, L., Paprzycki, M. (eds.) *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*, pp. 905–912. Polskie Towarzystwo Informatyczne, Institute of Electrical and Electronics Engineers (2017). <https://doi.org/10.15439/2017F449>
74. Hernes, M.: Consensus theory for cognitive agents' unstructured knowledge conflicts resolving in management information systems. In: Nguyen, N.T., Kowalczyk, R., Hernes, M. (eds.) *Transactions on Computational Collective Intelligence XXXII Lecture Notes in Computer Science*, vol. 11370, pp. 1–119. Springer, Berlin, Heidelberg (2019)
75. Nguyen, N.T.: Consensus systems for conflict solving in distributed systems. *Inf. Sci.* **147**(1–4), 91–122 (2002)
76. Nguyen, N.T.: *Advanced Methods for Inconsistent Knowledge Management*. Springer, London (2008)



Marcin Hernes is a researcher in Wrocław University of Economics. He is the author and coauthor of over 100 papers published in national and international publishing houses. Moreover, he is a participant of scientific national and international conferences. Marcin Hernes is a practitioner in the scope of management systems, coordination, and automation of production processes and multi-agent decision support systems.



Ngoc Thanh Nguyen is the author or coauthor of 5 monographs and more than 400 journal and conference papers. He serves as editor in chief of three international journals. He has given 20 plenary speeches for international conferences, and more than 40 invited lectures in many countries. In 2009, he was granted of title ACM Distinguished Scientist and Distinguished Visitor of IEEE.



Contents

50.1	Introduction	1003
50.2	Background and Research Motivation	1005
50.2.1	Research Purpose.....	1005
50.2.2	Research Procedure.....	1006
50.3	Fundamentals and Literature Review	1007
50.3.1	Cause of Failure.....	1007
50.3.2	Finite Element Method (FEM).....	1007
50.3.3	Fatigue Life Prediction.....	1007
50.3.4	Accelerated Life Testing (ALT).....	1008
50.3.5	Parameter Uncertainty.....	1009
50.3.6	Probabilistic Design and Reliability.....	1009
50.4	Modeling and Verification	1010
50.4.1	Configuration and Finite Element Modeling.....	1010
50.4.2	Verification of FEM.....	1014
50.4.3	Probabilistic FEM and ALT.....	1015
50.5	Result of Analysis	1015
50.5.1	Reliability of PBGA 316.....	1015
50.5.2	ALT of PBGA 316.....	1017
50.5.3	Combination of Reliability and ALT for PBGA 316.....	1018
50.5.4	Conclusion of Analysis.....	1020
50.6	Final Comments	1021
References	1022

Abstract

In the semiconductor industry, the study of reliability and the ability to predict temperature cycling fatigue life of electronic packaging are of significance. For that purpose, researchers and engineers frequently employ the finite element method (FEM) in their analyses. It is primarily a mechanics analysis tool that takes material properties, manufacturing processes, and environmental factors into consideration. Engineers also like to use FEM in their

design of electronic package, but frequently the term “reliability” they refer to only addresses the robustness of a particular design. It has little to do with probability and statistics. Meanwhile, in manufacturing factories of electronic products, including packaging, accelerated life testing (ALT) is carried out very often by quality engineers to find lives of a product in more severe environmental conditions than those of the field condition. Through regression analysis of the test result based on an empirical or semiempirical formula, the acceleration factor (AF) can be obtained for use in predicting service life of the product in field condition. Again, other than regression analysis, little probability and statistics are involved. By taking parameter uncertainties into consideration, this chapter demonstrates by an example that FEM, ALT, and AF can be combined to study the reliability of electronic packaging in which probability and statistics are applied.

Keywords

Electronic packaging · Parameter uncertainty · Life distribution · Probabilistic analysis · Statistical method · Reliability · Mechanics · Finite element method (FEM) · Accelerated life testing (ALT) · Acceleration factor (AF)

50.1 Introduction

The development of semiconductor in the past few decades has changed dramatically the way we live. Today, the semiconductor industry is still prosperous, and its development trend is that semiconductors are designed and manufactured smaller and smaller with their performances become better and better. Along with the development of semiconductor industry, electronic packaging industry has also been thriving. The purpose of packaging is to connect together and enclose appropriately semiconductor chips, components,

W.-F. Wu (✉) · Y.-A. Lu
Department of Mechanical Engineering, National Taiwan University,
Taipei, Taiwan
e-mail: wfwu@ntu.edu.tw

printed circuit boards, and other assemblies and modules to guarantee the performance of electronic devices and protects them from mechanical damage, cooling, radio frequency noise emission, and electrostatic discharge. Various kinds of failure have to be avoided during the manufacturing and product life cycle of a package. To that purpose, the reliability of a packaged electronic device or equipment has to be assured, and related research is needed [1–4]. Unlike the design of semiconductor that relies primarily on electronic and electrical engineers, electronic packaging is a major discipline within the field of mechanical engineering, in particular mechanics [5]. Principles such as dynamics, stress analysis, heat transfer, and fluid mechanics are employed extensively in the design and analysis of electronic packaging. Thousands of researches have been carried out in the last few decades under the general term of “reliability of electronic packaging.” However, most researches have nothing to do with probability and statistics. Researchers just use the word “reliability” to address that their packaging designs and products manufactured are strong and robust enough to encounter environmental stresses, avoid failures, and possess lives that are long enough.

A major issue among those encountered in the design and manufacture of electronic packaging is the mismatch of thermal conductivity among materials used in a package [6]. It may result in stress concentration, fracture, or warpage that affects the performance and eventually causes failure of the package. The life of package may not meet the need as well. To overcome the problem, analysis has to be carried out in the design stage of a new electronic package. Advanced researches are also needed accompanying with the development of smaller and smaller semiconductors. A powerful tool used in the design of package as well as the research of “reliability of electronic packaging” is the finite element analysis (FEA), also called finite element method (FEM) or finite element modeling (FEM) [7]. Based on mechanics principles, the method has been developed for more than 50 years and is very much matured today. Commercial software packages can be found in the market. A software package of this kind usually covers analyses of solid mechanics (including stress, strain, displacement, etc.), fluid mechanics, heat transfer, temperature fatigue life, and many others. Modules in consideration of uncertainty have been included in a few software packages in recent years but in a rather primitive stage. One purpose of the present chapter is to propose and illustrate by examples a more complete mechanics analysis of electronic packaging that employs FEM and takes into account parameter uncertainties of materials as well as environmental factors. By doing so, statistical analysis can be connected to mechanics analysis, and the result is truly meaningful under the term of “reliability of electronic packaging.”

In fact, the idea of connecting mechanics analysis with statistical analysis arose by the leading author of this chapter more than 10 years ago. The reason was that several of his acquaintances had been doing researches on “reliability of electronic packaging,” but in fact, their research contents had little to do with reliability in terms of probability and statistics. The Weibull plots they drew and explained for experimental data might even be incorrect. The leading author of this chapter considered that if researchers argued FEM can describe accurately the mechanics behavior of a package as compared with experimental data, it can also be used in carrying out numerical simulation of mechanics behavior of the package. Actually, the term “mechanics simulation” or “numerical simulation” based on FEM has long been used by researchers of “reliability of electronic packaging,” but the simulation is in an average sense of everything that takes no consideration of uncertainty. The “reliability” those researchers address just reflects the strength and robustness of the package without any quantitative measure. The purpose of this chapter is to carry out mechanics-based numerical simulation of electronic packaging by FEM and take into account uncertainties of the manufacturing process, environmental factors, and material properties. It can be considered a kind of numerical experiment instead of true experiments or tests. The latter are usually very expensive if quantitative reliability measure and statistically meaningful sample size are asked for. The simulation would save greatly the expense, and its result would resemble experimental or test result if appropriate uncertainties are measured and statistically analyzed in advance and then incorporated into the simulation. Even under the same environmental condition, the simulation result will be a set of data caused by the considered uncertain factors but not a single-value data. It resembles a sample in a statistical experiment. Statistical analyses can then be employed to treat the data for drawing conclusion related to not only mechanics but also quantitative reliability of the electronic package. It would achieve truly the analysis of “reliability of electronic packaging.”

Most of the following contents are abstracted from the master thesis of the second author under direction of the leading author of this chapter. It presents to a certain extent the applicability of the above idea with special emphasis put on accelerated life testing (ALT) which is frequently practiced in the electronic industry. The organization is as follows. In Sect. 50.2, the background and research motivation are addressed. In Sect. 50.3, literature and related theories are reviewed briefly. In addition to the introduction of FEM and ALT, parameter uncertainty and reliability is emphasized. They make this study different from others. In Sect. 50.4, the configuration of electronic package and the analytical model used in this study are introduced. Verification of the model is also presented. In Sect. 50.5, both results of the reliability and

ALT analyses taking into account parameter uncertainties are addressed, and conclusion is drawn based on results of the analysis. In Sect. 50.6, a few comments are made to finally conclude this chapter.

50.2 Background and Research Motivation

With the rise in both demands and popularity of electronic products such as smart phones and personal computers, electronic packaging, being a vital component in these products, has become the topic of numerous researches. In order to keep up with the current trend of low cost, lightweight, highly efficient electronic products, many technological advances in electronic packaging have been made in the past few years, such as the replacement of lead wires with solder balls and different board configurations accustomed to different needs. To reduce sizes of the electronic product, the geometry of package has also evolved from single chip package (SCP), multichip package (MCP), system in package (SIP), system on package (SOP), package on package (POP), system on chip (SOC), and wafer-level chip-scale package (WLCSP) to the so-called 2.5D and 3D packaging such as the most updated fan-out wafer-level packaging (FOWLP) [1, 3]. These advances present many new challenges for manufacturers and engineers, with a special emphasis on the reliability of package, which influences the life of a finished product. One of the most popular electronic packaging configurations is the ball grid array (BGA), with rows of solder balls transmitting signals on a square or rectangular chip. Depending on the manufacturing process, it can further be classified as PBGA (plastic BGA), CBGA (ceramic BGA), CCBGA (ceramic column BGA), CSP (chip scale package or μ BGA), and TBGA (tape BGA). The BGA has advantages of being very efficient for its small size and is the configuration of choice in many products. The main reliability concern of BGAs is to prevent fracture of solder balls and prolong their temperature-cycling fatigue lives. It is an important issue and has been emphasized in electronic industry under the name of “reliability of electronic packaging” [5, 8].

From mechanics point of view, the differences between coefficients of thermal expansion (CTE) of the solder balls and their surroundings result in creep strain accumulation in solder balls when subjected to temperature-cycling conditions. This could cause failure at the interfaces between solder balls and their connected chip after a number of cycles. To prevent failures in advance, design engineers frequently employ finite element method (FEM) for prototyping and parameter optimization of packages in the electronic packaging industry. Researchers [9, 10] have also used FEM in conjunction with fatigue formulas such as Coffin-Manson equation [11, 12] to relate damage parameters accumulated

per cycle with fatigue life. These damage parameters include creep strain and strain energy.

From quality point of view, the life of an electronic product operated in a normal environmental condition, temperature in particular, has to be assured when it is manufactured. To that purpose, temperature-cycling fatigue life tests in factories are frequently required by purchasers. However, under a normal environmental or field condition, the number of cycles that leads to product failure is significantly large, making experiments time consuming, expensive, and often impractical. In order to obtain the reliability properties of a given BGA package in a short amount of time, engineers often carry out accelerated life testing (ALT) that tests the product in more severe environmental conditions and then employ empirical or semiempirical formulas (or ALT models in short) to estimate fatigue life of the package when it is operated in the normal or field condition [13]. However, ALT presents its own problems in that many experiments are required to curve-fit the coefficients of ALT models and the accumulated time and cost are substantial. Also, the accuracy of the empirical models is questionable and needs to be recorrelated for different packages with different solder materials.

This research plans to connect the above two types of engineers, i.e., design engineers and quality engineers, by combining FEA and ALT for studying the reliability of a BGA. The proposed analysis can take advantage of the speed and low cost of FEM, and use it as an experimental tool to correlate the coefficients of ALT models. Current FEM tools such as ANSYS have the ability to run a probabilistic design system (PDS)-based simulation, introducing random parameter variations into the FEM model. This approach could be used to simulate the manufacturing variations and natural variation in material properties that are present in real-world experiments, allowing FEM to be used as a substitute for ALT under different environmental conditions. With the ability to perform a large number of tests under different environmental conditions at relatively low costs, the proposed method could generate more relevant data to curve-fit existing ALT models and produce results that are more accurate and closer to real-life conditions.

50.2.1 Research Purpose

Traditional researches on the topic of “reliability of electronic packaging” are of deterministic nature; however, real-life conditions involve parameter uncertainties that result in the fatigue life being a random nature and having a distribution. This research combines traditional FEM approaches with probabilistic design system (PDS) to investigate the effects of parameter uncertainties on fatigue life distribution and reliability of lead-free plastic ball grid arrays (PBGA) under

different environmental conditions. This study also aims to improve the accuracy of current accelerated life testing (ALT) models such as the Norris-Landzberg (N-L) formula [13] by using the probabilistic FEM model as a substitute for real-life experiments. With a large number of individuals generated by PDS for each of the test samples, we can perform numerous ALTs without expensive cost, resulting in a cheap and efficient prediction method. Fatigue life distribution and reliability of PBGAs under different temperature cycling conditions as well as their corresponding statistics are emphasized in this chapter.

50.2.2 Research Procedure

The proposed method of fatigue life prediction combining finite element analysis (FEA), accelerated life testing (ALT), and probabilistic design system (PDS) is summarized in Fig. 50.1 in which the research procedures are divided into four parts. The first part is preparation and FEM model

verification. With experimental data collected from reference [14], we can replicate a full temperature-cycling test using FEM. Using the exact geometric dimensions and material properties, we construct a representative finite element model subject to boundary conditions and temperature cycling profiles of the said experiment. The cycles are performed several times in order for the equivalent creep strain per cycle to stabilize. The final results are recorded and applied to the modified Coffin-Manson equation which relates the damage parameter per cycle with fatigue life of the PBGA assembly. We then introduce parameter uncertainties into the FEM model using PDS, with 3% coefficient of variation [15, 16] on selected geometric dimensions and material properties. This simulates a real-life sample of 30 tests, and the fatigue life distributions are recorded and studied. The results are then verified with experimental data to ensure the FEM model is an accurate representation of the actual test vehicle and can be used for other temperature-cycling tests.

The second part is the traditional deterministic fatigue analysis. The study uses the verified FEM model and per-

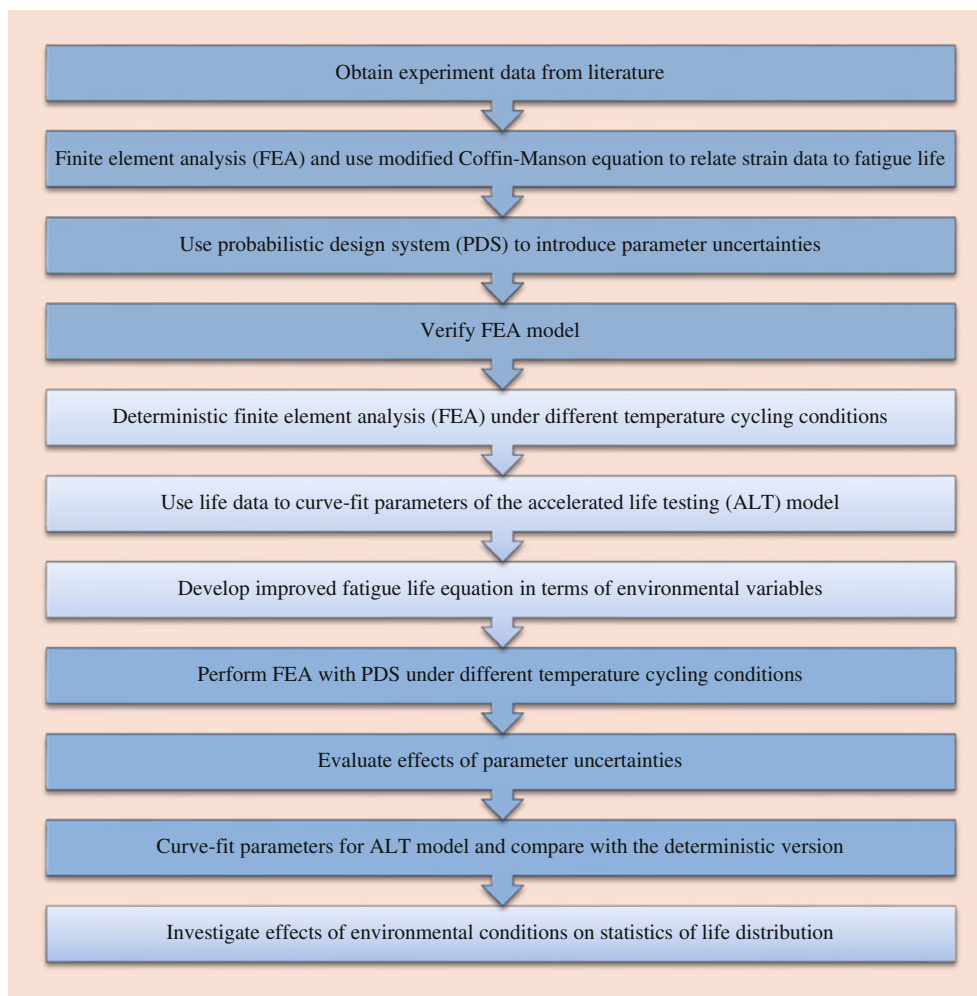


Fig. 50.1 Research procedure

forms various deterministic temperature cycling tests following JESD22-A104C of the JEDEC (Joint Electron Device Engineering Council) standards [17]. The fatigue life results are then used to curve-fit the N-L ALT empirical formula, and a deterministic version of the formula is devised. In the third part, we perform probabilistic FEM tests using PDS under each of the testing conditions. This creates samples of 30 per temperature-cycling condition resulting in a large set of data that we could use for further investigations, including the evaluation of the effect of each of the parameter uncertainties. A probabilistic N-L ALT formula is curve-fitted from the results, and its performance is compared with its deterministic counterpart. The fourth and final part of this chapter is to investigate the effect of environmental condition on the statistical quantities of fatigue life. The standard deviation, Weibull scale, shape, and threshold parameters are discussed in particular, and an attempt to summarize their behavior under different temperature loading is carried out.

While the above procedures were proposed and carried out by the second author of this chapter, it should be mentioned that they have been reorganized a little by the leading author in the following sections and subsections.

50.3 Fundamentals and Literature Review

This section gives an overview of fundamental theories used in this research and, in the meantime, reviews related works done by previous researchers on the subject of “reliability of electronic packaging.”

50.3.1 Cause of Failure

Temperature-cycling fatigue has always been a major source of failure in electronic components and products, and many researches have been carried out to investigate its contributing factors. Darveaux et al. [18] were among those earlier researchers who discovered that differences of coefficient of thermal expansion (CTE) among materials are the main cause of failure of electronic components and products under temperature-cycling conditions. They showed that a small amount of CTE difference is sufficient in producing instantaneous thermal strain, and the strain will accumulate under temperature-cycling conditions to cause failure eventually.

The above temperature-cycling fatigue failure generally arises from the propagation of a small crack in solder balls in which materials containing lead were frequently used before. Spurred by a 2003 European Union directive known as RoHS (Restriction of Hazardous Substances), lead-free materials have been required in recent years. In fact, as discovered by Kim et al. [19], lead-free solder has a lower crack propagation rate as compared to that with lead. The above situations

led to many more studies of reliability of electronic assemblies using solder material such as various configurations of Sn-Ag-Cu (SAC) solder. Among them, Meilunas et al. [20] have studied the reliability of both lead and lead-free solder joints under accelerated life tests. In their study, failures were determined and observed by X-ray, cross sectioning, and dye penetration, and the results showed that lead-free solders indeed have better performance against fatigue.

50.3.2 Finite Element Method (FEM)

Aside from actual temperature cycling fatigue tests, finite element method (FEM) has gained popularity for the simulation of mechanics behavior of electronic packages. A wide variety of assemblies has been the subject of research using FEM, notably Syed [21] who employed previous framework of SnPb modeling on SnAgCu solder alloys, with the intention of formulating a life prediction model that can be used in early design stages of chip development. Accumulated creep strain-based and creep strain density-based models were developed, but the prediction errors were up to 25% in certain cases. Pang and his associates [22] have done a collection of studies on lead-free solder with one particular example carried out by Che et al. [14] that focused on Sn-3.8Ag-0.7Cu solder joint electronic assemblies such as PBGA 316, PQFP208, PQFP176, and TSSOP48. The FEM as well as experiment were used to determine the mean time to failure (MTTF) and Weibull parameters against temperature cycles. The vehicle dimensions and material properties are the basis of this study and experimental data being a source of FEM verification.

To save computation time, submodeling techniques are commonly used in FEM. Lai and Wang [23] have once compared the calculation time between full and submodeling for flip-chip packages. They pointed out that computation time and cost can be saved using submodeling technique and found the solder ball on outer diagonal of the chip is the first to fail. Different types of reduced model have also been used to save computation time. They include slice model or strip model, quadrant model, and octant model with various degrees of accuracy and efficiency [24]. One thing we need to keep in mind while using reduced and submodeling techniques is whether they have the same failure location and mechanisms as the full model, otherwise the simplification will lead to errors and contradictions.

50.3.3 Fatigue Life Prediction

Fatigue is defined as the process of permanent deformation caused by continuously repeated loading. The loading is often of a cyclic nature and could be significantly smaller

than the yielding stress of the material. During the process of fatigue failure, the material properties gradually change, and cracks and creases are formed within the material. These imperfections tend to increase both in size and number as the cyclic loading continues, leading to total failure of the material or structure. In this chapter, we are concerned with the temperature fatigue failure of electronic packages, which is the result of thermal stress induced by temperature-cycling condition. These thermal stresses are caused by the mismatch of coefficient of thermal expansion at the interfaces of different material in the electronic package.

In the study of electronic packaging reliability, it is crucial to have a way to predict the fatigue life of components subject to temperature cycles. Many different fatigue life formulas have been developed and tested by researchers. One of the most widely used formulas is the Coffin-Manson equation, also called Coffin-Manson model [12, 13], which uses plastic strain amplitude as the damage parameter for correlating with fatigue life. The Coffin-Manson model is used mainly in the prediction of temperature cycling fatigue life of lead solders, but modification for use on lead-free solders has also produced reasonable results. Other models such as the Goldmann model [25] focuses on the relationship between fatigue life and geometry/material properties of the package, and Darveaux et al. [18] tackled the problem using energy-based methods, correlating inelastic strain energy density with fatigue life. Most of these models and methods are constructed using experimental data and are empirical or semiempirical formulas that have certain degrees of variation due to human and experimental errors. Further enhancements and modifications are needed.

In this study, the following modified Coffin-Manson fatigue life prediction formula is used:

$$N = \frac{1}{2} \left(\frac{\Delta\gamma}{2\varepsilon_f} \right)^{\frac{1}{C_f}} \quad (50.1)$$

where N is the fatigue life, ε_f is a ductility coefficient, C_f is the fatigue ductility exponent, and $\Delta\gamma$ is the shear strain range also called strain amplitude per loading cycle. The last quantity is related to the von Mises strain range $\Delta\varepsilon$ as follows

$$\Delta\gamma = \sqrt{3}\Delta\varepsilon \quad (50.2)$$

Since the shear strain range thus obtained is in an indirect manner, it is also called equivalent shear strain range or equivalent creep strain in short.

50.3.4 Accelerated Life Testing (ALT)

As mentioned before, another fatigue life prediction method used frequently in the electronics industry is accelerated life

testing (ALT). It tests electronic components including packages under harsh environmental conditions to obtain their failure times which are then correlated to components' lives under normal conditions through acceleration factor (AF) derived based on physical principles. Norris and Landzberg [13] used temperature cycling parameters such as cyclic frequency, maximum temperature, and temperature difference to deduce an empirical formula that calculates the AF between two different environmental conditions. The coefficients of the N-L formula or N-L model were curve-fitted using lead solders, but as lead-free solder gains popularity, investigations of the corresponding coefficients were carried out at later stages. These include Pan et al. [26], but with limited success as experimental results and predictions were found to have low correlation as those pointed out by Salmela et al. [27] and Zhang and Clech [28]. New models and modifications were proposed with new parameters added, such as the works of Dauksher [29] and Vasudevan and Fan [30], with a special emphasis on experimental confirmation. Jong et al. [31] discovered that the size of BGA package and other design parameters also have a profound effect on the AF, leading to an even more complicated model. For all above results, there is still a large discrepancy among researchers, which deserves further evaluation and improvements.

There are two major groups of accelerated life testing: One is to increase the operating frequency of the test subject, and the other to increase the intensity of the test loading. Raising the number of repetitive operation needs to be carried out carefully, if the frequency becomes too high, the operating environment may change and result in higher load intensity. On the other hand, while increasing experiment loading, one should keep in mind that the failure mode and mechanism should be consistent with normal loading for the experiment to be meaningful. For electronics, increasing environmental loading such as temperature or vibration is a common method to shorten the testing time. After obtaining life data from accelerated life testing, one can use the abovementioned acceleration factor (AF) to gain information of the test vehicle under normal operating condition. The definition of AF is the ratio between the fatigue lives of two environments, as shown below:

$$AF = \frac{N_1}{N_2} \quad (50.3)$$

where N_1 and N_2 are fatigue lives under two different environmental conditions. Frequently, N_1 is considered the life under normal or field condition, and N_2 the life under accelerated or more severe condition; the expression becomes

$$AF = \frac{N_{\text{field}}}{N_{\text{test}}} \quad (50.4)$$

It results in that AF is always greater than one.

Although ALT is frequently carried out in the electronics industry, it may not be known to researchers and/or engineers who employ FEM in the design and implementation of electronic packaging. Only a few researchers have made the connections. Among them, Norris and Landzberg are two pioneers who proposed the following Norris-Landzberg model [13]:

$$AF = \left(\frac{f_{\text{field}}}{f_{\text{test}}} \right)^m \left(\frac{\Delta T_{\text{test}}}{\Delta T_{\text{field}}} \right)^n \exp \left[\frac{E_a}{k} \left(\frac{1}{T_{\text{max,field}}} - \frac{1}{T_{\text{max,test}}} \right) \right] \quad (50.5)$$

where f represents cyclic frequency of temperature change, ΔT represents the temperature range of a cycle, T_{max} is the maximum temperature in a cycle, E_a is the activation energy, k is Boltzmann's constant, and m and n are empirical constants to be determined from test data. The model is also called N-L model in this study for simplicity. It should be noted that while a few parameters in the above expressions bear physical meanings, some are to be determined from test data. It is the nature of ALT-data treatment in real practices.

50.3.5 Parameter Uncertainty

As stated before, many engineers and researchers engaging in designs and/or researches related to “reliability of electronic packaging” placed their emphases on mechanics analysis of packages. They frequently employ FEM and empirical or semiempirical fatigue prediction formulas in their analyses. Although Weibull probability distribution has been adopted by a small amount of people in fitting experimental life data to obtain the mean or characteristic life of a package, the “reliability” they mentioned really has little to do with probability and statistics. The mechanics behavior most people addressed is also the average trend of a package. With regard to ALT employed frequently in factories of electronic products, although more life data than those from FEM are usually obtained, only the average life is considered in most analyses. The acceleration factor (AF) which engineers and/or researchers try to find from test result is also in an average sense in most cases. However, uncertainty exists in the real world, and if one wants the term “reliability of electronic packaging” to reflect quantitative reliability, one can take into account uncertainties of parameters in mechanics analysis. The major difference of this study from others is that FEM and ALT are employed at the same time and uncertainty is taken into consideration as well. By doing so, life distribution rather than life in its average sense can be obtained, and quantitative reliability can be addressed and predicted. If mechanics researchers assert their analyses based on FEM reflect the true behavior of electronic

packaging, FEM can be used for numerical simulation tool to obtain lives of a package subject to different environmental conditions. Moreover, if parameter uncertainty is considered, the simulation result will be life distribution rather than just a single-value life under a certain environmental condition. Using FEM as a tool to carry out numerical test in place of real ALT is another innovative part of this study.

Uncertainties may arise during both the design and manufacturing stages of electronic packaging. The effect of parameter uncertainty on the prediction of fatigue life of electronic assemblies has been studied and discussed previously. In a study carried out by Mercado and Sarihan [32] on plastic ball grid array (PBGA) assemblies, uncertainties of parameters including substrate thickness, solder height, and chip thickness were addressed and investigated using FEM. Perkins and Sitaraman [33] have also employed FEM to run a full factorial design of simulations (DOS) in which variations of parameters such as substrate size, CTE mismatch between components, substrate/board thickness, and joint pitch were discussed. In a later study, Wu and Barker [34] found the geometric variation caused by imperfections in the manufacturing process will result in discrete life distribution, but the effects of material properties are still open to further investigations.

In summary, both FEM and ALT can be used for fatigue life prediction of electronic components including packages when they are subject to a certain environmental condition, in particular temperature-cycling condition in this study. Parameter uncertainties would make the predicted life exhibit probability distribution which is related to quantitative reliability.

50.3.6 Probabilistic Design and Reliability

Traditional engineering analyses including FEM are performed under many different assumptions and simplifications concerning the dimensions and material properties. In the real world, uncertainties are present in all of these parameters, some of which may have significant effects on the evaluation results. It is therefore worth considering those effects. In fact, many software packages used in engineering design and/or analysis have developed probabilistic modules to treat uncertainties of parameters in recent years. However, they are usually in rather primitive stages. The module has been named “probabilistic design system” (PDS) in some commercial software packages. It works by assuming some parameters involved in the analysis follow normal distribution, randomly generating a large amount of sample points for each uncertain parameter, carrying out finite element analyses, handling the results by descriptive statistics, and discussing the effects of parameter uncertainties on the results. The present study follows similar procedures but treats the

simulation results in a more detailed manner. In particular, the results are related to quantitative reliability which has seldom been seen before. Since the simulation is also considered as numerical substitution of real ALT, its results are treated as those obtained from ALT. The overall concept is also new, at least to the writers.

Unlike traditional and deterministic FEM, parameter uncertainties cause an electronic package to have life distribution rather than a single-value life under the same environmental condition and in consideration of the same material properties. Under this circumstance, reliability analysis based on probability and statistics can be carried out to treat those simulated life data. Unfortunately, many engineers are not familiar with probability and statistics and can only perform mechanics analyses using traditional and deterministic FEM. A few of them sometimes use the following Weibull probability density function to fit experimental life data for the sake of comparing with their FEM results.

$$f(t) = \frac{\beta}{\theta} \left(\frac{t-\gamma}{\theta} \right)^{\beta-1} \exp \left[- \left(\frac{t-\gamma}{\theta} \right)^\beta \right], \quad (50.6)$$

$$\beta > 0, \theta > 0, \gamma \geq 0, t \geq 0.$$

in which t is the life or fatigue life of a package in this study, θ is a scale parameter which is also called characteristic life, β is the shape parameter, and γ is a threshold parameter. The mean and variance of the above distribution are

$$\mu = \gamma + \theta \Gamma \left(1 + \frac{1}{\beta} \right) \quad (50.7)$$

and

$$\sigma^2 = \theta^2 \left\{ \Gamma \left(1 + \frac{2}{\beta} \right) - \left[\Gamma \left(1 + \frac{1}{\beta} \right) \right]^2 \right\} \quad (50.8)$$

respectively, in which $\Gamma(\cdot)$ is the gamma function. The quantity of Eq. (50.7) is frequently called the mean time to failure (MTTF) in reliability engineering. The above three-parameter Weibull becomes two-parameter Weibull probability density function if the threshold parameter is 0. It is interesting to note that, other than normal probability distribution, Weibull probability distribution is frequently used in various engineering fields without knowing its reason by most users. The leading author of this chapter believes, other than its versatility in fitting all kinds of data, the fact that Weibull was trained to be an engineer and eventually became an outstanding professor makes the difference. In fact, Weibull distribution is also used frequently in treating test data of ALT. However, many engineers misinterpret the characteristic life as the mean life.

50.4 Modeling and Verification

This section will introduce configuration of the studied electronic package and its material properties. One emphasis is finite element modeling of the package and its verification. After the analytical result being verified, the FEM can then be used for performing numerical simulation of ALT, which will be addressed in the last part of this section.

50.4.1 Configuration and Finite Element Modeling

When running mechanics analysis concerning stress and strain under certain loading and boundary conditions, analytical methods are only applicable to subjects with simple geometries. However, engineering problems often involve real-life parts and components that are very complex and would be impossible to solve analytically. Therefore, a numerical analysis method that could break down the problems into smaller and manageable subproblems such as the finite element method (FEM) will be needed for such engineering needs. FEM divides an irregular geometry into many connected simple polygons to approximate the result of a certain loading by solving the relevant partial differential equations. In the electronic industry, FEM is often used as a tool to prototype chip design, since it can save time and money otherwise necessary for experiments and manufacturing. In this study, we utilized FEM to conduct our research on lead-free electronic assembly under temperature-cycling conditions to obtain the accumulated creep strain data needed for fatigue life prediction.

In order to execute the methodology proposed in this study, we first need to construct a finite element model that can be verified as an acceptable representation of real-life experiment. While many different types of lead-free electronic assemblies are in use today, it is essential to use a package that has experimental data available for calibration of our model. We chose the plastic ball grid array with 316 inputs/outputs (PBGA 316) presented in [14] as the basis of our model, and its results as our benchmark for accuracy confirmation. The schematic diagram of the package is shown in Fig. 50.2, and the layout of 316 solder balls is shown in Fig. 50.3. The bottom layer of the package is an FR4 PCB (printed circuit board) with solder balls on top of it. Above the solder balls lies a BT (bismaleimide triazine) substrate, and on top of the substrate is a mold with the die embedded in the center. The package has basically a square configuration, and its dimensions are summarized in Table 50.1.

The material properties are listed in Tables 50.2 and 50.3. Since we are dealing with creep behavior of electronic components under repeated temperature cycles, the material

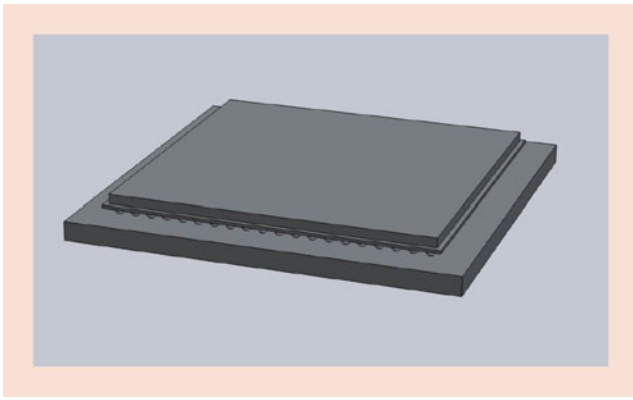


Fig. 50.2 Schematic diagram of PBGA 316

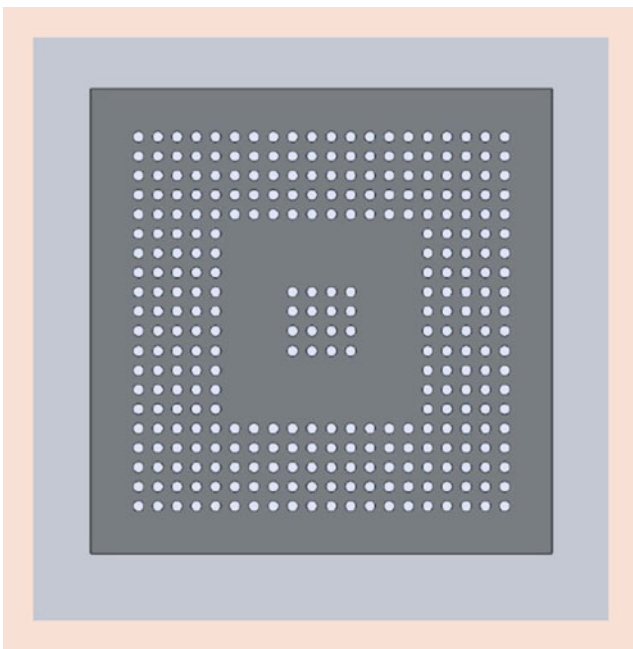


Fig. 50.3 Layout of solder balls in PBGA 316

Table 50.1 Dimensions of PBGA 316

Solder ball number	316
Pitch (mm)	1.27
Solder ball radius (mm)	0.20
Solder ball height (mm)	0.58
Die length (mm)	7.60
FR4 PCB length (mm)	30.5
BT substrate length (mm)	26.2
Mold length (mm)	25.4
Die thickness (mm)	0.25
FR4 PCB thickness (mm)	1.50
BT substrate thickness (mm)	0.25
Mold thickness (mm)	0.75

Table 50.2 Material properties of PBGA 316 [14]

Material	Modulus (GPa)	Poisson's ratio	CTE (ppm/°C)
Solder	See table in [14]	0.35	24.5
FR4 PCB (in plane)	20	0.28	18
FR4 PCB (out of plane)	9.8	0.11	50
Copper	155.17	0.34	See Table 50.3
BT substrate (in plane)	26	0.39	15
BT substrate (out of plane)	11	0.11	52
Die	See Table 50.3	0.278	See Table 50.3
Adhesive	7.38	0.3	52
Mold	16	0.25	15

Table 50.3 Time-dependent material properties [14]

Temperature	-40 °C	25 °C	50 °C	125 °C
Solder modulus (GPa)	54.43	41.73	36.84	22.19
Copper CTE (ppm/°C)	15.3	16.4	16.7	17.3
Die modulus (GPa)	192.1	191	190.6	190
Die CTE (ppm/°C)	1.5	2.6	2.8	3.1

properties that change with temperature have to be handled carefully with corresponding values entered at different stages of the analysis.

With regard to this package, although FEM can simulate a given environmental loading with accuracy, it is still near impossible to fully represent all of the physical processes that are present in a certain analysis, in the case of this study, the strain distribution of PBGA under temperature cycling. Therefore, it is essential that we make the following assumptions in our analysis using FEM: (1) The model is stress-free at room temperature (25 °C), and there are no residual stress caused by the manufacturing process. (2) The displacement on the symmetrical plane along the symmetric direction is 0. (3) The material properties of all of the components concerned are homogenous and isotropic. (4) All contact surfaces within the model are in full contact, and there are no impurities or voids inside the electronic packages. (5) Ignore the popcorn effect caused by humidity and temperature change. (6) Ignore heat conduction, convection, and radiation effects. (7) The inner parts of the model are of the same temperature as the outside temperature. That is to say, the system is under thermal equilibrium at all instants during the analysis and the thermal field only changes with time. It is interesting to point out that the above idealization through assumptions justifies to a certain degree that our later analysis taking into account uncertainties is meaningful.

Carrying out finite element analysis on a complex structure such as the PBGA array concerned in this study presents

itself with many difficulties. One of the obstacles to overcome is the long computational time required to evaluate the creep behavior of each of the 316 solder balls. With the need to repeat the analysis many times under different design parameters, the task quickly becomes unmanageable. It is therefore essential to make simplifications to the model to shorten the calculation time; fortunately, the symmetry of assembly means many different methods are available to improve the efficiency of the analysis. One of them is the so-called reduced modeling. It is a method used to take advantage of symmetry of the test subject. In the example of our study, the PBGA 316 assembly can be divided into different numbers of subsections. One popular method among researchers is the one quadrant reduced model [9] shown in Fig. 50.4. It is an effective method to obtain accurate result with a reasonable reduction in computational time, but with the amount of repetitions needed for this research, an even more efficient method is necessary. The slice model [31] shown in Fig. 50.5 is one such method, and it has been shown that the sacrifice in accuracy is within the acceptable range. Therefore, it is used as a basis of the rest of this study. The slice is split along the diagonal axis of the package and has the width of one-half of the pitch of the solder balls.

The technique of submodeling is also adopted in this study. It is a combination of reduced modeling and full modeling, in order to achieve better accuracy under a shorter amount of time [35]. While using the submodeling technique, the global model is usually meshed with a coarse mesh, giving a rough idea of the global displacement of the assembly. The data is then fed into various submodels at areas around

those focusing points such as the location of maximum stress or the point of failure. These submodels are meshed using a much finer mesh, as they are where maximum accuracy is desired. As opposed to the large global model, the smaller submodel will make fine meshes much less time consuming while producing satisfactory results, which is a reasonable compromise between accuracy and efficiency.

While using reduced modeling techniques such as the three-dimensional slice models, it is essential to apply the correct boundary and loading conditions in order to give an accurate representation of real-life testing. The conditions for our model follow that of the slice model found in literature [31]. We develop a 1/2 symmetric slice model along the diagonal of the package as that shown in Fig. 50.6 in which the origin is set at the bottom of the symmetric center, having x points to the long direction and y points upward. The boundary conditions of the reduced model are also shown in Fig. 50.6. It can be seen that all finite element nodes at the center and sliced surface of the package are set to be “symmetric” while those at the bottom are set to be “fixed.” The fixed condition means the displacement of those points in all directions are 0, preventing rigid body motion that would occur if being set otherwise, giving the model the freedom to deform and warp freely. On the surface at half of the ball pitch in the negative z -direction is where most difficulties are found, as it is neither a free surface nor a true symmetry plane. In order for this surface to move freely along the z -axis and to prevent out-of-plane stress, we assign it with generalized plane strain condition. The remaining surfaces not mentioned are all assumed to be free surfaces.

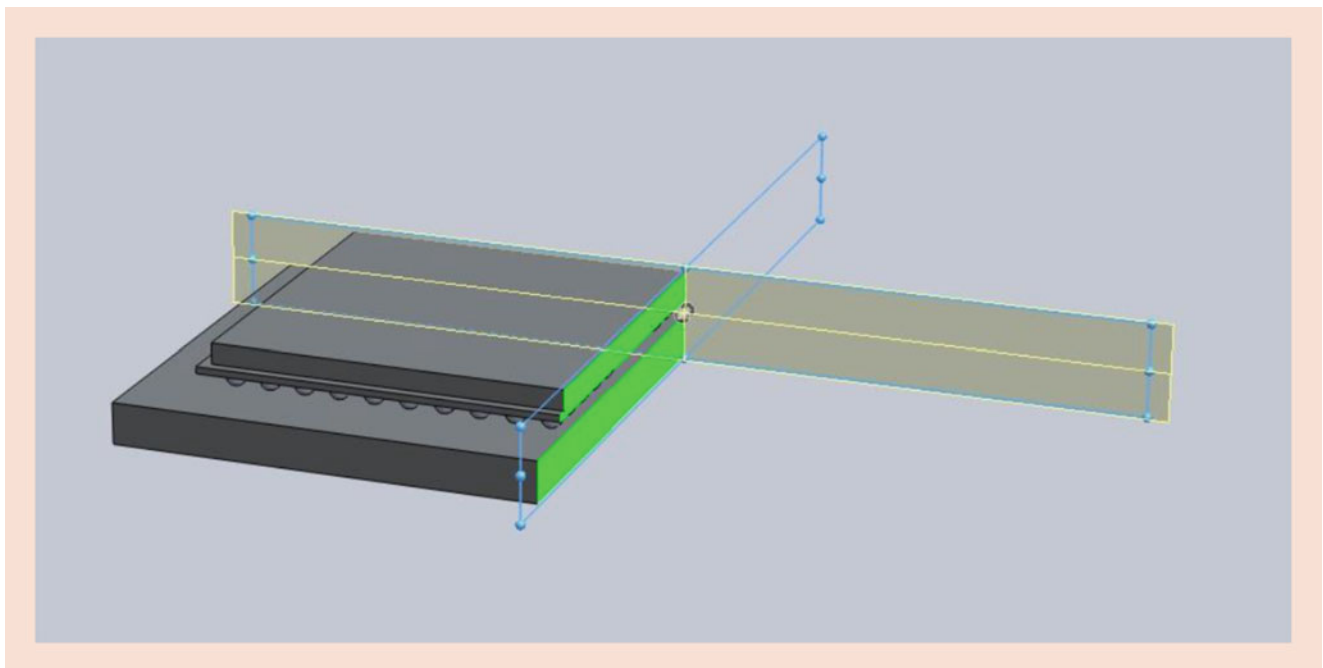


Fig. 50.4 One quadrant reduced model

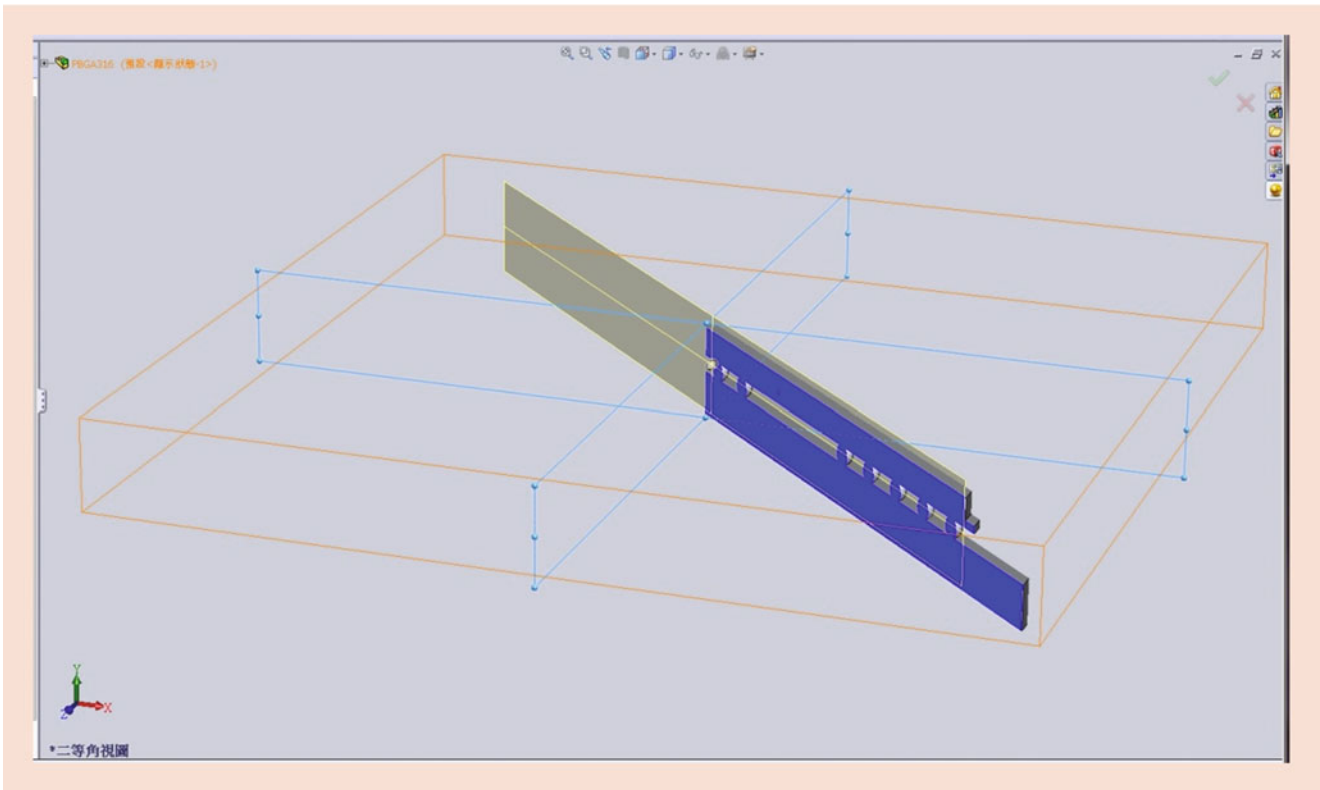


Fig. 50.5 Slice model

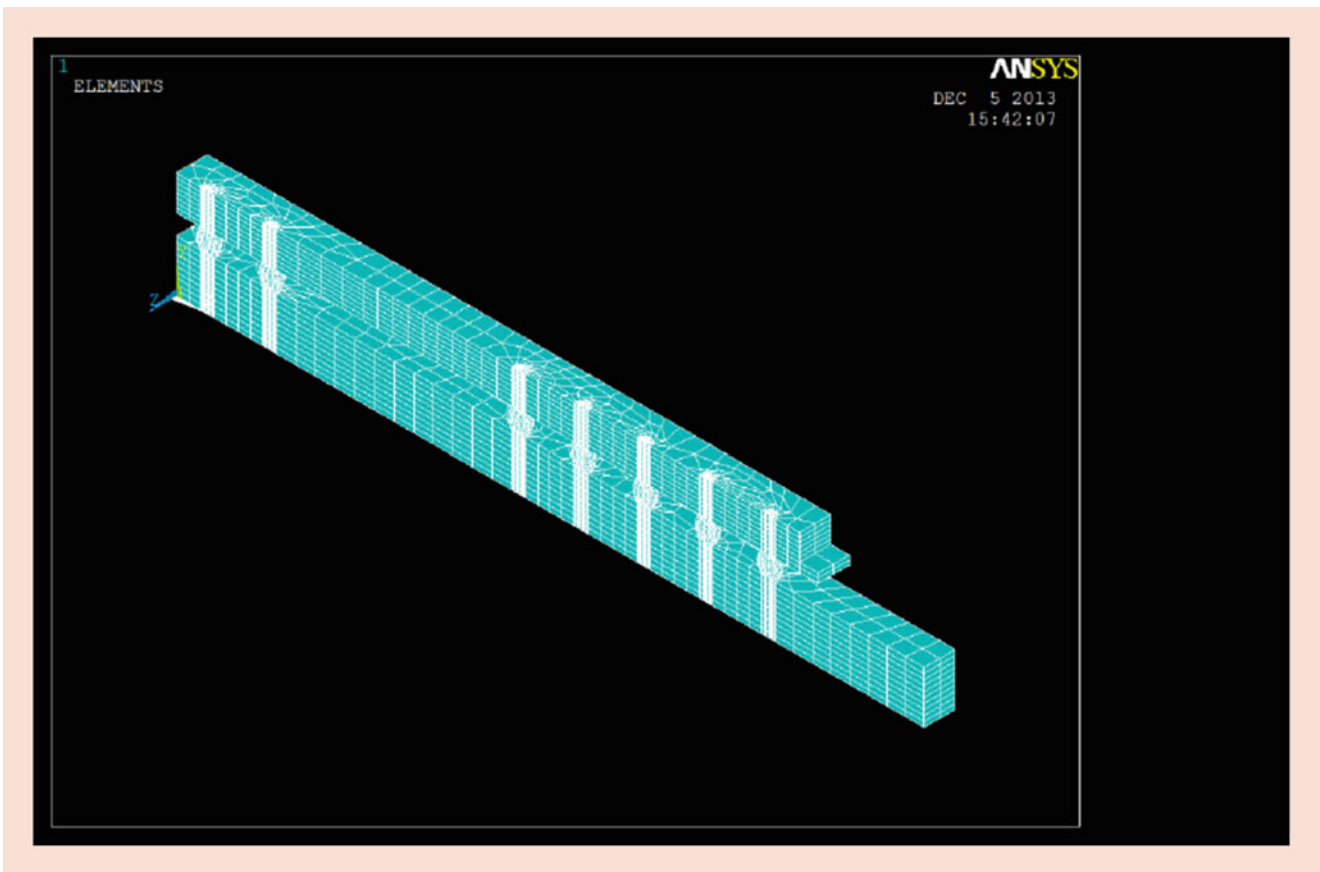


Fig. 50.6 Reduced finite element model of PBGA 316

The temperature-cycling conditions used in the finite element analysis follow those of JEDEC testing standard JESD22-A104C, a set of guidelines devised for testing of electronic reliability as a result of CTE mismatch between materials under different temperatures. Starting from room temperature, the loading conditions are described by four load steps: two dwells, one at the highest cycle temperature and one at the lowest, and two temperature ramps in between the dwells. The cycle can be represented visually using the graph shown in Fig. 50.7. In this study, since FEM is proposed as numerical substitution for ALT, six test conditions covering a wide variety of temperatures and cycle times as those shown in Table 50.4 are numerically studied. It should be noted that values in the last two columns of Table 50.4 can be used for converting fatigue life from cycles to hours. In later sections, these loading cycles will be used in FEM analysis to determine the AF of the PBGA 316 assembly once the model is verified by experiment results. For the time being, we chose Test A [14] as the basis of our comparison and verification.

During the finite element analysis, meshing is a crucial stage. Finer meshes result in more accurate result but are computationally expensive. In order to carry out a large set of simulation runs like our proposed method, it is important to strike the right balance for the size of the mesh. In this study, we examine different mesh density. Through many trial and errors with different meshing options, it has been found that the length of the simulation time mostly depends on the size

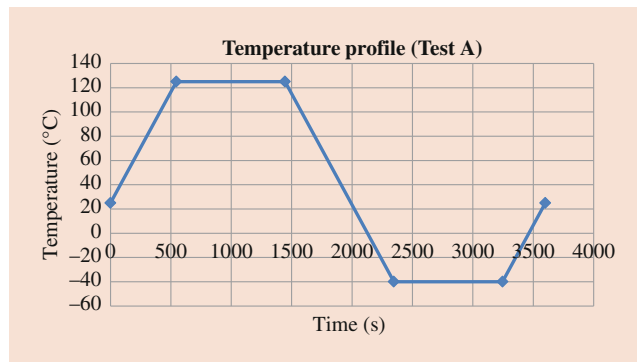


Fig. 50.7 Temperature profile of Test A

Table 50.4 Temperature-cycling conditions

Test condition	Temperature range (°C)	Ramp rate (°C/min)	Dwell time (min)	Cycle time (min/cycle)	Frequency (Cycle/day)
Test A	-40 ~ 125	11	15	60	24
Test B	0 ~ 100	6.67	15	60	24
Test C	-40 ~ 85	13.89	1	20	72
Test D	-40 ~ 150	9.5	10	60	24
Test E	-55 ~ 85	14	5	30	48
Test F	-55 ~ 150	14.64	1	30	48

of the mesh covering the solder balls, which is expected as the majority of the calculation resource is spent on modeling of the solder ball's creep behavior.

In order to evaluate the reliability of PBGA 316 assembly using FEM, we need to correlate the numerical result of FEM with experimental fatigue life of the package. As stated before, we can use the Coffin-Manson model for our life prediction, using maximum equivalent creep strain among all elements per cycle as the damage parameter.

50.4.2 Verification of FEM

The result of the above finite element analysis indicates the location of maximum equivalent creep strain occurs at the interface (with PCB) of the outmost solder ball in the central 4×4 cluster, which is consistent with literature data. A graph of the development of equivalent creep strain along with time is shown in Fig. 50.8. It is found that value of the equivalent creep strain per cycle stabilizes after three cycles. Therefore, the equivalent creep strain range after the fourth cycle is used in simulation afterward. The fatigue life of the assembly can then be found by applying the Coffin-Manson relationship of Eqs. (50.1) and (50.2), in which the fatigue ductility coefficient is taken to be 0.325 [16, 27, 36]. The value is found to be 2746 temperature cycles or 2746 h since a temperature cycle is 1 h (Fig. 50.7). The result is consistent with experimental data listed in Table 50.5 by Che et al. [14]. It is interesting to note that, although small samples are usually obtained (as that of Table 50.5 which is a small sample of 5), engineers like to fit their experimental or test data by Weibull probability distribution. However, they usually do not address quantitative reliability based on fitted distribution of the fatigue life and sometime misinterpret the characteristic life as the mean life. The test data thus fitted by Che et al. [14] has a Weibull slope $\beta = 7.69$ and a characteristic life $\theta = 2917$ corresponding to an MTTF of 2742 cycles calculated from Eq. (50.7). It is also interesting to note the five test lives in Table 50.5 exhibit a considerable scatter which is attributed to uncertainties of parameters mentioned previously and emphasized in this study.

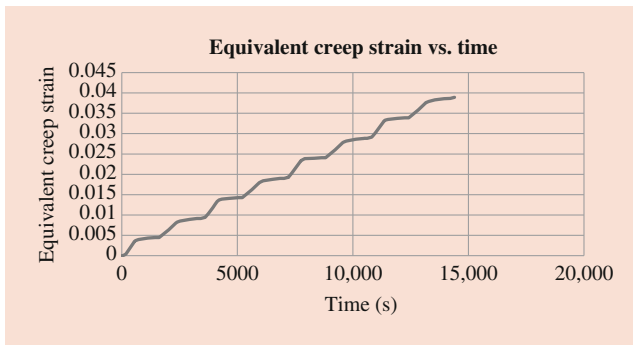


Fig. 50.8 Development of equivalent creep strain

Table 50.5 Experimental data [14]

Test number	Fatigue life (cycles)
1	2131
2	2745
3	2892
4	2972
5	3052
Average	2758

50.4.3 Probabilistic FEM and ALT

As stated before, ALT is frequently employed in the electronics industry, but owing to the long test-time and the expensive cost, its results under different environmental conditions are usually small samples from statistics point of view. If researchers and engineers assert their numerical analysis based on FEM can capture the mechanics behavior and predict the temperature-cycling fatigue life of electronic packaging accurately, the analysis can be used for substitution of ALT to save time and cost. Moreover, if parameter uncertainties involved in the analysis are considered, the result of FEM would not be a single-value fatigue life but a sample of fatigue life under a particular environmental condition. Thus, not only ALT under different environmental conditions can be substituted by numerical simulation based on FEM but also the results of fatigue lives exhibit distributions which are related to quantitative reliability and can be treated by any statistical methods. It can also bridge the gap between design engineers and reliability engineers.

It has been shown that uncertainties in geometric parameters and material properties have a significant effect on the temperature cycling fatigue life of electronic assemblies [33]. To construct an accurate simulation of real-life conditions in this study, we assume the radius and height of the solder ball, the thickness of the BT substrate, and Young's Moduli and CTEs of both the mold and the BT substrate in x , z plane are random variables having normal distribution. They are assumed to have mean values shown in Tables 50.1 and 50.2, and a 3% coefficient of variation in each of the assumed

Table 50.6 Simulation results

	Test A	Test B	Test C	Test D	Test E	Test F
Average (h)	2947	9216	12,522	1482	10,687	1501
Standard deviation (h)	571	1093	3981	292	2853	188

random parameters. A lot of simulations have been carried out, and some of the results will be reported in the following section.

50.5 Result of Analysis

As mentioned preciously, introducing parameter uncertainties into the analysis of FEM can form a methodology that generates a large set of temperature fatigue life data of a package but with less cost than that of real-life test. By doing so, the procedure is combined with ALT under different temperature-cycling environments, and the results can further be correlated to the acceleration factors (AFs). Also stated before, the temperature profiles used in this study follow the regulation of JESD22-A104C and are shown in Table 50.4. The choices of temperature profiles include the original condition used for verification purposes (Test A) and others that explore different maximum temperatures, different temperature differences, different cycling frequency, and even different dwell times and ramp times. The variety of these profiles gives more flexibility while constructing the ALT model, as many parameters could be considered while performing regression analyses for coefficients. The analysis (also called simulation) is performed on the PBGA 316 model for each of the six temperature profiles listed in Table 50.4. In the analysis, the maximum equivalent strain is found and applied to the Coffin-Manson relationship to produce fatigue life in cycles. It can be converted to life in hours if values of the last two columns in Table 50.4 are taken into consideration.

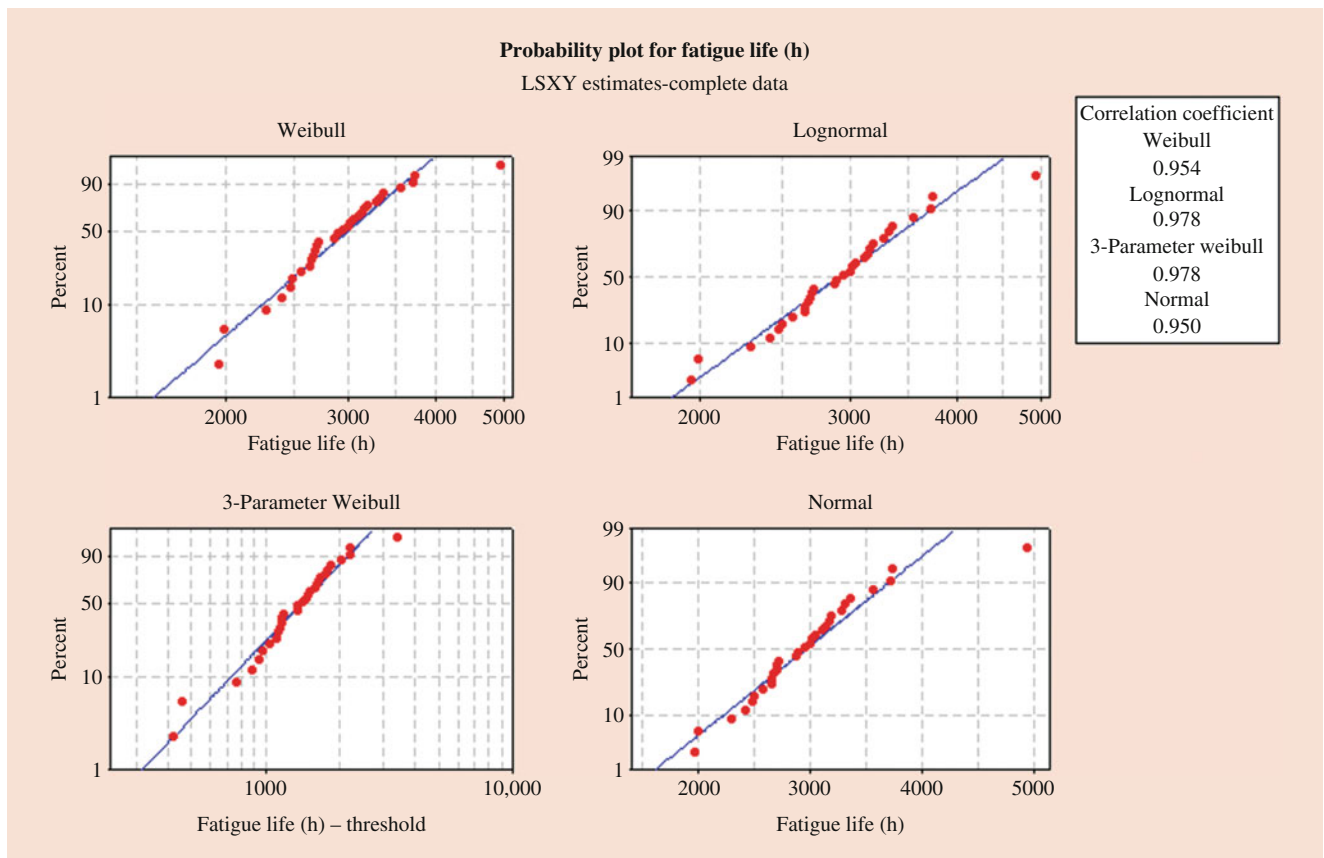
Some results of this study will be introduced in the following three subsections. The content of the first subsection is related to quantitative reliability, the second subsection emphasizes more on ALT, and the third subsection tries a preliminary study in combining the previous two.

50.5.1 Reliability of PBGA 316

In this study, the simulation is carried out 30 times for each of the six test conditions resulting in 180 fatigue life data in total. After being transformed into hours, the mean values and standard deviations of the results are shown in Table 50.6. The Pearson correlation coefficients between the fatigue life and each of the assumed random parameters are shown in Table 50.7. We can see from the table that all variables

Table 50.7 Pearson correlation coefficients of variables

Testcondition	Solder radius	Mold thickness	BT substrate thickness	Mold modulus	BT substrate modulus	Mold CTE	BT substrate CTE
Test A	0.104	-0.141	0.557	-0.016	0.383	0.146	0.323
Test B	0.445	-0.158	0.310	0.046	0.123	0.072	-0.089
Test C	-0.131	-0.384	-0.126	-0.258	-0.156	-0.541	-0.046
Test D	-0.023	-0.161	0.351	0.053	0.009	0.139	0.396
Test E	0.231	-0.346	0.375	-0.129	-0.056	-0.119	-0.096
Test F	-0.007	0.205	0.032	0.081	0.178	-0.099	0.167
Average	0.103	-0.164	0.250	-0.037	0.080	-0.067	0.109

**Fig. 50.9** Comparison among distributions (test A)

are moderately correlated to fatigue life of the assembly, with the thickness of the BT substrate showing the most obvious positive trend. This trend suggests that increasing the thickness of the BT substrate can extend the fatigue life of PBGA assemblies, the likely cause being a reduction of warpage with thicker substrates, which leads to a decrease in stress at the interfaces of the components. It is interesting to point out that this kind of observation coincides with conclusions made by other researchers based primarily on mechanics analysis and, in fact, it should be like this.

The probabilistic method proposed in this study results in multiple data for each test condition or temperature profile, which can be considered random fatigue life having a proba-

bility distribution. The distribution can be used for reliability evaluation and life prediction of the studied package such as the failure rate and mean time to failure (MTTF). As stated previously, traditional research often uses Weibull distribution as the default distribution for describing temperature fatigue life. However, it has been shown to not always be the case in real-life experiments. In our study, within concepts of Anderson-Darling test and the least squared method, we fit our life data-sets by normal, lognormal, Weibull, and 3-parameter Weibull distributions with correlation coefficient used as our indicator. It results in probability plots such as those shown in Fig. 50.9 for Test A. Probability plots for other tests conditions are neglected herein to save space. The

summarized result in Table 50.8 shows that 5 out of the 6 tests are best fitted with 3-parameter Weibull distribution, which is consistent with industry knowledge as the threshold term of 3-parameter Weibull distribution implies a range of time where no failure will occur. It is interesting to point out that results such as in Fig. 50.9 and Table 50.8 can be used for quantitative reliability evaluation of electronic packaging but much of the time ignored by engineers who just want to obtain characteristic life and/or MTTF of the package through fitting test data primarily by Weibull probability distribution. In our study, when we use 3-parameter Weibull distribution to describe all sets of data, the corresponding parameters are found and shown in Table 50.9, in which the MTTF and standard deviation of life of the package under each test condition are shown in the last two rows in hours. When comparing with those data in Table 50.6, the MTTFs are very close to mean values of the simulated data and the standard deviations are also within 10% difference from those simulated data except for Test B. One can also see that all shape parameters are larger than 1 indicating failure rate of the system increases with time, which is expected in an electronic assembly. The threshold parameter provides valuable information for manufacturers and engineers as the failure-free time can be used for guidelines for design and experimental purposes. The probability density function, reliability function, and failure rate function of the tested package can be plotted accordingly as those shown in Fig. 50.10 for Test A. The latter two quantities are frequently emphasized in reliability engineering.

50.5.2 ALT of PBGA 316

As mentioned before, a major difference of our study from others is that ALT is carried out numerically but not

experimentally. Although the concept of applying Norris-Landzberg model (or N-L model in brief) of Eq. (50.5) to find AF factor from the ALT result may not be new, the experimental or analytical data points are usually very few. Previous studies such as [27, 36] were based primarily on deterministic analyses with only a single data or an average data point considered for each of the temperature-cycling conditions. With the probabilistic approach, we have instead 30 data points for each temperature-cycling condition and a total of 180 data points, giving us more flexibility when performing regression analysis in search of parametric values such as m , n , and E_a/k in Eq. (50.5).

In our study, the temperature-cycling condition of Test A is considered the test condition in Eq. (50.5) with each of the other conditions being considered a particular field condition. After values of m , n , and E_a/k are evaluated, the package life under each field condition is estimated from the test condition and compared with FEM simulation result. As mentioned previously, the FEM simulation is proposed to substitute real ALT and can also be called numerical ALT.

Two different approaches are employed with regard to the regression analysis of those simulation data. The first approach uses all of the PDS data for curve fitting, with all 180 data points contributing in obtaining the unknown variables in the AF model of Eq. (50.5). The second approach uses average values of each of the conditions as the data set in the regression analysis. The latter still considers the effects of all 180 simulation results but is much simpler. The first approach results in $m = 0.36$, $n = 2.43$, and $E_a/k = 940.96$. We then set Test A as our test condition and consider the remaining 5 temperature-cycling conditions as different field conditions for verifying the accuracy of Eq. (50.5). Following what is usually done in real ALT and also for the sake of simplicity, only the mean value of the 30 data points for each condition is considered at this stage. The predicted temperature fatigue lives based on the simulation result of Test A and Eq. (50.5) are compared with simulation results of other tests, all in average sense. The prediction errors for each of the condition are calculated and shown in Table 50.10. Although the errors are comparably large, they are compatible with and even better than those found by Salmela and his associates when comparing their simulation result with test data [27, 36]. For the second approach, the best fitted parametric values are $m = 0.14$, $n = 1.04$, and

Table 50.8 Correlation coefficients for different distributions

	Test A	Test B	Test C	Test D	Test E	Test F
Normal	0.950	0.957	0.866	0.993	0.969	0.963
Lognormal	0.978	0.963	0.934	0.986	0.991	0.961
Weibull	0.954	0.908	0.859	0.988	0.968	0.960
3-P Weibull	0.978	0.986	0.984	0.992	0.993	0.979

Table 50.9 Parameters of 3-P Weibull

	Test A	Test B	Test C	Test D	Test E	Test F
3-P Weibull scale	1581.79	1624.28	4054.47	1034.48	6246.04	433.59
3-P Weibull shape	2.84	1.22	1.09	3.37	2.01	1.80
3-P Weibull threshold	1532.47	7724.19	8536.95	552.70	5166.68	1120.34
MTTF (h)	2942	9246	12,461	1482	10,701	1506
Standard deviation (h)	535	1254	3583	305	2897	172

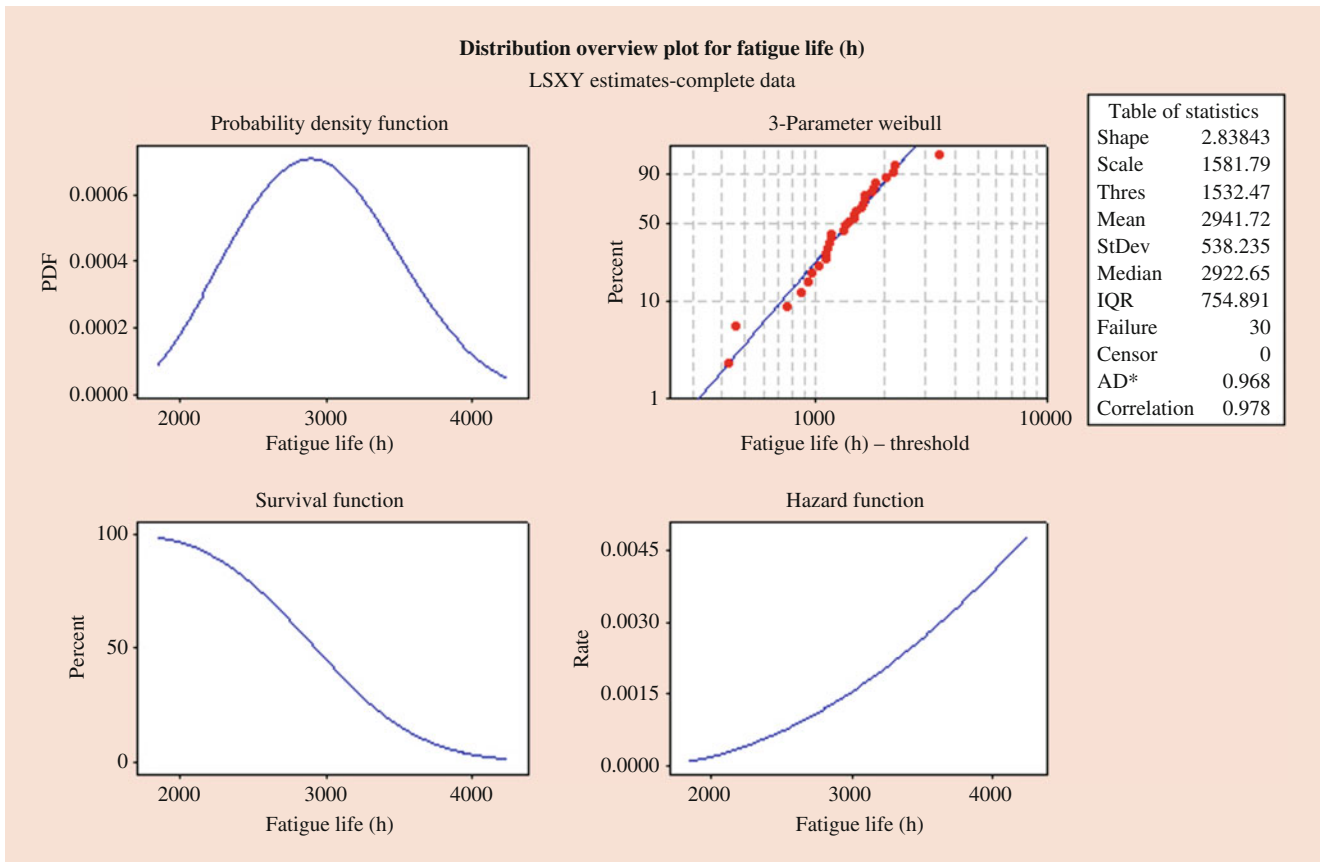


Fig. 50.10 Reliability of the package (test A)

Table 50.10 Prediction errors of approach 1

Condition	Error
Test A	–
Test B	28%
Test C	20%
Test D	16%
Test E	23%
Test F	12%
Average	20%

Table 50.11 Prediction results and errors of approach 2

Condition	Fatigue life (h)	Predicted fatigue life (h)	Error
Test A	2947	–	–
Test B	9216	9183	0.36%
Test C	12,522	12,763	1.92%
Test D	1476	1478	0.14%
Test E	10,687	10,737	0.47%
Test F	1501	1501	0.00%
Average			0.58%

$E_a/k = 3662.06$, and the prediction results become those shown in Table 50.11. Although the approach is simpler, the results are surprisingly better than those of the first approach. It needs further study to clarify the ambiguity.

50.5.3 Combination of Reliability and ALT for PBGA 316

It has been emphasized several times in the present chapter that traditional study on the topic of “reliability of electronic packaging” in fact has little to do with probability, statistics, and reliability. Although accelerated life testing (ALT)

has been carried out frequently in manufacturing factories of electronic products including packaging, the afterward analyses usually focus only on finding acceleration factor (AF) between the test condition and the field condition in an average sense. Other than regression analysis, no probability and statistics are involved in the analyses. If the finite element analysis is considered an appropriate method by researchers and engineers in studying “reliability of electronic packaging,” it can be extended to generate more data than those obtained from real test under a given condition by taking parameter uncertainties into consideration. Probability theory and statistical methods can then be applied to treat data

obtained through numerical simulation and draw conclusion other than that of the traditional analysis. The above two subsections present results in this study with one emphasizing in reliability of PBGA 316 and the other emphasizing in numerical ALT of PBGA 316 together with its afterward analysis. The present subsection tries to combine the above two topics.

In addition to evaluating parametric values and predicting the average life in practical ALT, we originally considered predicting the life distribution of the studied PBGA 316 under a specific temperature-cycling condition for results similar to those of Fig. 50.9. Since it is too complicated, we turn to predicting statistical properties of the life distribution. To make it simple in engineering applications, we assume the statistical property other than the mean value of life has the following linear relation with the environmental parameters:

$$S.P. = af + bT_{max} + c\Delta T + d \tag{50.9}$$

in which S. P. indicates the statistical property and the other parameters are defined in the statement after Eq. (50.5), with their meanings being seen in Fig. 50.7 and Table 50.4. Based on results of Tables 50.6 and 50.9, coefficients *a*, *b*, *c*, and *d* can be found by regression analysis, and the results are shown in Table 50.12. The corresponding prediction

values and errors (in parenthesis) are summarized in Table 50.13 and Fig. 50.11. From the results, we can see that the predictions of the threshold parameter of the 3-parameter Weibull distribution are quite accurate with an average error of less than 10%, showing that the linear regression method is applicable for this application. However, predictions for other parameters are not that good, and the method also deviates from the original purpose of ALT. Therefore, we seek for further improvements.

The practical ALT is intended for predicting life of an electronic product in a particular filed condition using life data of the product tested usually in more severely environmental conditions. However, as stated previously, only the average trend of life but not life distribution is considered. Since many life data have been generated through PDS of FEM, the present study tries to go a step further investigating the effect of temperature-cycling condition on life distribution of the studied PBGA 316. It is hoped that once life distributions under two or more test conditions are obtained, and their relation being established, the quantitative reliability of package under any other environmental conditions can be evaluated or estimated. This is in fact a problem of studying the relation between two degraded random processes and their corresponding lives, each under its own environmental condition. The problem is too complicated for engineers. Therefore, we focus on the AF of Eq. (50.5) based on Norris-Landzberg model and again consider Test A as our test condition and the others as field conditions. Only a few statistical properties are considered for simplicity. We want to find out whether the statistical properties are related to environmental conditions following Eq. (50.5). Using the data in Tables 50.6 and 50.9 and the AFs in the average sense, the respective results for *m*, *n*, and *E_a/k* are found and listed in Table 50.14. The prediction values and the errors (in parenthesis) concerning each of the statistical properties are summarized in Table 50.15 and Fig. 50.12. We can see that the shape parameter of the 3-parameter Weibull distribution shows the best prediction results with the average error less than 5%. This could mean the trend in failure rate of the assembly is highly dependent on the temperature-cycling condition and can be predicted with the

Table 50.12 Prediction coefficients using linear regression

Statistical property of life	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>
Standard deviation	34.60	−40.61	5.24	4008.08
Coefficient of variation	0.00067	−0.0034	0.0018	0.30
3-P Weibull scale	−31.32	−151.28	70.42	10,442.18
3-P Weibull shape	−0.041	−0.029	0.031	2.24
3-P Weibull threshold	90.61	44.14	−104.21	11,461.24

Table 50.13 Predicted statistical properties using liner regression

Statistical property of life	Test A	Test B	Test C	Test D	Test E	Test F
Standard deviation	621.4 (8.84%)	1298.4 (18.84%)	3698.2 (7.11%)	−263.6 (190.33%)	2945.8 (3.26%)	644.8 (243.52%)
Coefficient of variation	0.18138 (6.38%)	0.14908 (25.76%)	0.27834 (12.46%)	0.14088 (28.74%)	0.28956 (8.48%)	0.18426 (47.38%)
3-P Weibull scale	2431.8 (53.74%)	1630.8 (0.40%)	4153.4 (2.44%)	416.8 (59.71%)	5960.6 (4.57%)	721.6 (66.43%)
3-P Weibull shape	2.7721 (2.32%)	1.4756 (20.97%)	0.7393 (32.18%)	2.8246 (16.15%)	2.1802 (8.21%)	2.3167 (26.68%)
3-P Weibull threshold	1987.9 (29.72%)	7645.4 (1.02%)	8732.7 (2.29%)	490.4 (11.27%)	4998.3 (3.26%)	1104.8 (1.39%)

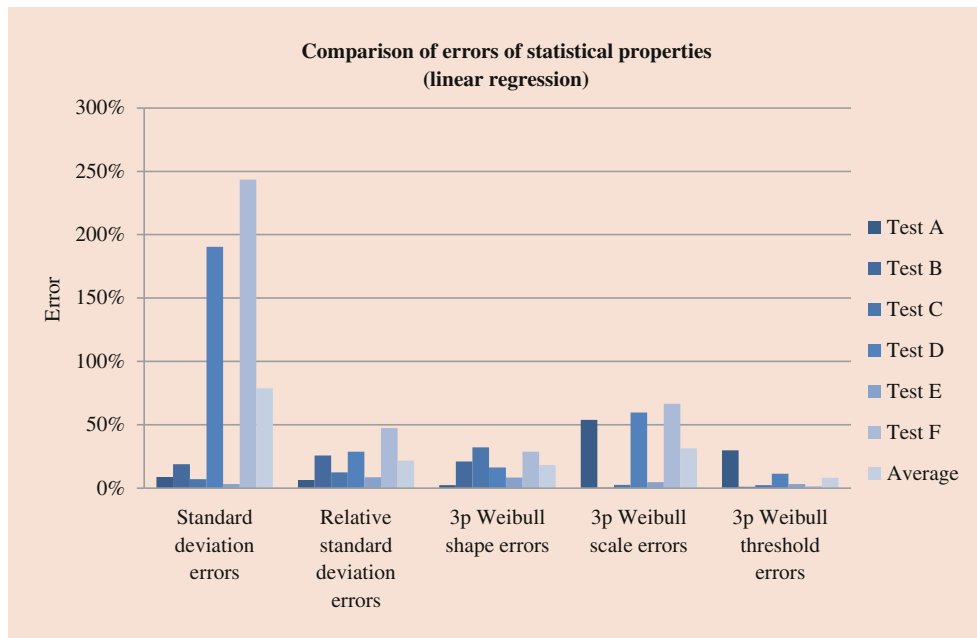


Fig. 50.11 Prediction errors of statistical property using linear regression

Table 50.14 AF coefficients for statistical properties

Statistical property of life	m	n	E_a/k
Standard deviation	-0.48	-0.15	5309.34
Coefficient of variation	-0.30	-2.22	3898.83
3-P Weibull scale	-0.98	-1.91	6665.62
3-P Weibull shape	-0.86	-2.48	2389.43
3-P Weibull threshold	1.13	1.32	5556.47

knowledge of the environmental parameters. It is interesting to see that while the linear regression method did a good job predicting the threshold parameter of the distribution, the AF of Eq. (50.5) did poorly and shows the inability of the N-L AF model to handle low-temperature cycling of Test C and Test E, each with a significantly higher error than the other tests. Aside from the shape parameter, the other predictions of statistical properties prove to be less than satisfying, which implies the environmental variables used for constructing the AF model may be unsuitable for these terms and further studies need to be carried out.

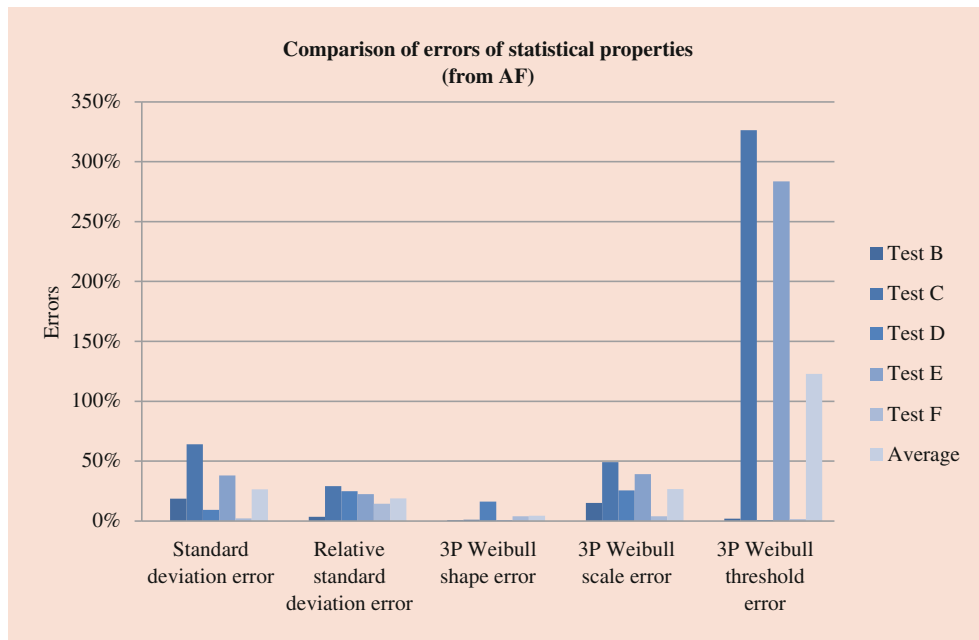
50.5.4 Conclusion of Analysis

The study proposes a method of fatigue life prediction of electronic assemblies that improves on previous models by implementing probabilistic concepts into FEM analysis.

Starting with the construction and verification of an FEM model, the method uses the model as an experimental tool instead of a design tool to perform accelerated life testing, bridging the gap between design engineers and test engineers. Working with the AF based on N-L model, the deterministic approach (which is not reported in this chapter) showed an inability to predict the fatigue lives of PBGA316 under lower temperature-cycling condition with a maximum temperature of 85 °C. With the introduction of parameter uncertainties in solder balls, mold, and BT substrate, a fatigue life distribution was found for each of the six environmental conditions. The height of the BT substrate has the most obvious influence on temperature-cycling fatigue life of the assembly, and it was found that the 3-parameter Weibull distribution can be used to describe the sets of data. The AF obtained using PDS showed a vast improvement over the deterministic model, with average errors less than 0.6% and the ability to handle lower temperature-cycling condition. Finally, while investigating the effect different temperature cycling conditions have on the distribution of thermal fatigue life, predictions of statistical properties such as standard variation and Weibull coefficients were carried out. It has been found that the threshold parameter of the 3-P Weibull distribution can be accurately obtained using a simple linear regression method, while the shape parameter of the 3-P Weibull distribution can be predicted using a modified AF based on A-L model. This implies the dependency of temperature-cycling fatigue life distribution on environment parameters including the maximum temperature, cycling frequency, and the temperature range in a temperature cycle.

Table 50.15 Predicted statistical properties from AF

Statistical property of life	Test A	Test B	Test C	Test D	Test E	Test F
Standard deviation	–	1295.88 (18.61%)	1432.76 (64.01%)	265.03 (9.18%)	1771.37 (37.91%)	191.96 (2.27%)
Coefficient of variation	–	0.12281 (3.60%)	0.22509 (29.21%)	0.14856 (24.86%)	0.32677 (22.41%)	0.14302 (14.39%)
3-P Weibull scale	–	1867.47 (14.97%)	2060.40 (49.18%)	769.66 (25.60%)	3806.52 (39.06%)	451.15 (4.05%)
3-P Weibull shape	–	1.22601 (0.51%)	1.07948 (0.98%)	2.82383 (16.17%)	2.02940 (0.73%)	1.87341 (4.06%)
3-P Weibull threshold	–	7565.71 (2.10%)	36,403.78 (326.41%)	557.41 (0.94%)	19,824.20 (283.71%)	1103.51 (122.90%)

**Fig. 50.12** Prediction errors of statistical property from AF

50.6 Final Comments

The application of FEM to study the quantitative reliability of electronic packaging caused by parameter uncertainties is meaningful and has been demonstrated in this study. The uncertainties may include material, process, and environmental factors since they all are parameters involved in the analysis of FEM. The environmental factors, in particular, are emphasized in practical ALTs in which one tries to find the relationship between lives of a product subjected to two or more different environmental conditions. Once the relationship is established, it can be used for predicting lives of the product subjected to various environmental conditions. The present study proposes to employ FEM as a kind of numerical ALT to substitute real ALT or part of real ALT in order to save test cost. It has been demonstrated the AF of ALT thus obtained can be used to predict the MTTF of the studied PBGA 316 subjected to a given temperature-

cycling condition rather accurately. However, for the purpose of reliability prediction, life distribution or higher-order statistics of life in addition to MTTF have to be found, and the relationship between life distributions or their statistical properties of two different environmental conditions has to be established. The demonstration in this study unfortunately is not very successful. One major reason is that engineers and/or engineering students are usually not very familiar with probability and statistics, which limits their analyses in that respect. It is believed that research can be carried out one step further in the future based on the concept proposed and demonstrated in the present study. The knowledge of probability and statistics is, of course, a plus to this kind of study.

It is interesting to point out that “uncertainty” addressed in this chapter has also been noticed in two recent articles by Wei et al. [37, 38]. Although the propagation of uncertainty is emphasized therein, it still justifies to a certain degree the concept of this chapter.

References

- Lau, J.H.: Recent advances and trends in fan-out wafer/panel-level packaging. *ASME J. Electron. Packag.* **141**(4), 040801 (2019)
- Lau, J.H.: Recent advances and new trends in flip chip technology. *ASME J. Electron. Packag.* **138**(3), 1–23 (2016)
- Lau, J.H.: *Fan-Out Wafer-Level Packaging*. Springer, Singapore (2018)
- Lau, J.H.: *3D IC Integration and Packaging*. McGraw-Hill Book Company, New York (2016)
- Viswanadham, P.: *Essentials of Electronic Packaging: A Multidisciplinary Approach*, Electronic Packaging Book Series. ASME, New York (2011)
- Broughton, J., Smet, V., Tummala, R.R., Joshi, Y.K.: Review of thermal packaging technologies for automotive power electronics for traction purposes. *ASME J. Electron. Packag.* **140**(4), 040801 (2018)
- Shen, Y., Zhang, L., Zhu, W., Zhou, J., Fan, X.: Finite-element analysis and experimental test for a capped-die flip chip package design. *IEEE Trans. Compon. Packag. Manuf. Technol.* **6**(9), 1308–1316 (2016)
- Shao, J., Zhang, H., Chen, B.: Experimental study on the reliability of PBGA electronic packaging under shock loading. *Electronics*. **8**(3), 279 (2019)
- Zulkifli, M.N., Jamal, Z.A.Z., Quadir, G.A.: Temperature cycling analysis for ball grid array package using finite element analysis. *Microelectron. Int.* **28**(1), 17–28 (2011)
- Jagarkal, S.G., Hossain, M.M., Agonafer, D., Lulu, M., Reh, S.: Design optimization and reliability of PWB level electronic package. In: *IEEE Ninth Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*, Vol. 2, pp. 368–376 (2004)
- Coffin Jr., L.F.: A study of the effects of cyclic thermal stresses on a ductile metal. *Trans. ASME*. **76**, 931–950 (1954)
- Manson, S.S.: *Thermal Stress and Low-Cycle Fatigue*. McGraw-Hill, New York (1966)
- Norris, K.C., Landzberg, A.H.: Reliability of controlled collapse interconnections. *IBM J. Res. Dev.* **13**(3), 266–271 (1969)
- Che, F.X., Pang, J.H., Xiong, B.S., Xu, L., Low, T.H.: Lead free solder joint reliability characterization for PBGA, PQFP and TSSOP assemblies. In: *Proceedings of IEEE 55th Electronic Components and Technology Conference*, pp. 916–921 (2005)
- Su, C.Y.: Probabilistic design and reliability analysis of flip-chip chip scale packages under accelerated environmental conditions. Ph.D. Dissertation, National Taiwan University, Taipei (2012)
- Chou, P.L.: Investigation of fatigue life of wafer-level chip-scale packages under thermal cycling conditions by acceleration models. M.S. Thesis, National Taiwan University, Taipei (2012)
- JEDEC Solid State Technology Association: *JESD22-A104C: Temperature Cycling* (2005)
- Darveaux, R., Turlik, I., Hwang, L.T., Reisman, A.: Thermal stress analysis of a multichip package design. *IEEE Trans. Compon. Hybrids Manuf. Technol.* **12**(4), 663–672 (1989)
- Kim, D.H., Elenius, P., Barrett, S.: Solder joint reliability and characteristics of deformation and crack growth of Sn-Ag-Cu versus eutectic Sn-Pb on a WLP in a thermal cycling test. *IEEE Trans. Electron. Packag. Manuf.* **25**(2), 84–90 (2002)
- Meilunas, M., Primavera, A., Dunford, S.O.: Reliability and failure analysis of lead-free solder joints. In: *Proceedings of the IPC Annual Meeting* (2002)
- Syed, A.: Accumulated creep strain and energy density based thermal fatigue life prediction models for SnAgCu solder joints. *Proceedings of IEEE 54th. Electronic Components and Technology Conference*, Vol. 1, pp. 737–746 (2004)
- Pang, J.H.L.: *Lead Free Solder: Mechanics and Reliability*. Springer, New York (2011)
- Lai, Y.S., Wang, T.H.: Verification of submodeling technique in thermomechanical reliability assessment of flip-chip package assembly. *Microelectron. Reliab.* **45**(3), 575–582 (2005)
- Pang, J.H.L., Low, T.H., Xiong, B.S., Che, F.X.: Design for reliability (DFR) methodology for electronic packaging assemblies. In: *Proceedings of the 5th Electronics Packaging Technology Conference (EPTC 2003)*, pp. 470–478 (2003)
- Lall, P., Shirgaokar, A., Arunachalam, D.: Norris–Landzberg acceleration factors and Goldmann constants for SAC305 lead-free electronics. *ASME Journal of Electronic Packaging*. **134**(3), 031008 (2012)
- Pan, N., Henshall, G.A., Billaut, F., Dai, S., Strum, N.J., Lewis, R., Benedetto, E., Rayner, J.: An acceleration model for Sn-Ag-Cu solder joint reliability under various thermal cycle conditions. In: *Proceedings of the SMTA International Conference*, pp. 876–883 (2005)
- Salmela, O., Andersson, K., Sarkka, J., Tammenmaa, M.: Reliability analysis of some ceramic lead-free solder attachments. In: *Proceedings of the SMTA Pan Pacific Conference*, pp. 161–169 (2005)
- Zhang, R., Clech, J.P.: Applicability of various Pb-free solder joint acceleration factor models. In: *Proceedings of the SMTA International Conference* (2006)
- Dauksher, W.: A second-level SAC solder-joint fatigue-life prediction methodology. *IEEE Trans. Device Mater. Reliab.* **8**(1), 168–173 (2008)
- Vasudevan, V., Fan, X.: An acceleration model for lead-free (SAC) solder joint reliability under thermal cycling. In: *Proceedings of the 2008 Electronic Components and Technology Conference*, IEEE, pp. 139–145 (2008)
- Jong, W.R., Chen, S.C., Tsai, H.C., Chiu, C.C., Chang, H.T.: The geometrical effects of bumps on the fatigue life of flip-chip packages by Taguchi method. *J. Reinf. Plast. Compos.* **25**(1), 99–114 (2006)
- Mercado, L.L., Sarihan, V.: Predictive design of flip-chip PBGA for high reliability and low cost. In: *Proceedings of the 2008 Electronic Components and Technology Conference*, IEEE, pp. 1111–1115 (1999)
- Perkins, A., Sitaraman, S.K.: Universal fatigue life prediction equation for ceramic ball grid array (CBGA) packages. *Microelectron. Reliab.* **47**(12), 2260–2274 (2007)
- Wu, M.L., Barker, D.: Rapid assessment of BGA life under vibration and bending, and influence of input parameter uncertainties. *Microelectron. Reliab.* **50**(1), 140–148 (2010)
- Cheng, H.C., Yu, C.Y., Chen, W.H.: An effective thermal-mechanical modeling methodology for large-scale area array typed packages. *Comput. Model. Eng. Sci.* **7**(1), 1–17 (2005)
- Salmela, O.: Acceleration factors for lead-free solder materials. *IEEE Trans. Compon. Packag. Technol.* **30**(4), 161–169 (2005)
- Wei, H.P., Yang, Y.H., Han, B.: Stacking yield prediction of package-on-package assembly using advanced uncertainty propagation analysis: part I stochastic model development. *ASME J. Electron. Packag.* **142**(1), 011001 (2020)
- Wei, H.P., Yang, Y.H., Han, B.: Stacking yield prediction of package-on-package assembly using advanced uncertainty propagation analysis: part II implementation of stochastic model. *ASME J. Electron. Packag.* **142**(1), 011002 (2020)



Wen-Fang Wu received his BS degree from National Taiwan University (NTU) in 1977 and PhD from University of Illinois at Urbana-Champaign in 1985. He had worked at Florida Atlantic University and Columbia University before joining NTU in 1988. He is now a professor of Mechanical Engineering and Industrial Engineering. His research interests include vibration, reliability engineering, and probabilistic risk assessment.



Yi-An Lu received BS and MS degrees from National Taiwan University (NTU) in 2011 and 2014, respectively. Although being educated as an engineer, he is interested in the composition and performance of popular music and has organized a double chorus band named “crispy” in Taiwan. The band has been nominated for the Golden Melody Award of Taiwan in 2018.



Qingchuan He , Wen-Hua Chen , and Jun Pan

Contents

51.1	Introduction	1025
51.2	Types of ALT	1027
51.3	Life Distribution	1028
51.3.1	Exponential Distribution.....	1028
51.3.2	Lognormal Distribution.....	1028
51.3.3	Weibull Distribution.....	1028
51.4	Life-Stress Models	1029
51.4.1	Exponential Model.....	1029
51.4.2	Power Model.....	1030
51.5	Statistical Analysis of ALT Data	1031
51.6	Design of ALT Plans	1035
51.6.1	Method for Planning Single-Stress CSALTs.....	1035
51.6.2	Method for Planning Multiple-Stresses CSALTs.....	1036
51.6.3	Design of SSALT Plans.....	1036
51.7	Some Pitfalls of Accelerated Testing	1037
51.8	Summary	1037
	References	1037

background and motivations for using accelerated testing and classifies the reliability tests. Section 51.2 provides the basic concepts and factors, which should be taken into account in planning and conducting ALTs. Sections 51.3 and 51.4 provide brief descriptions of specific applications of statistical models including the life distribution and the life-stress relationship. Section 51.5 illustrates an approach for analyzing ALT data. The graphical and numerical methods are discussed for fitting an ALT model to data and for assessing its fit. Section 51.6 describes the research development of the methods for planning optimal ALT with location-scale distribution. Section 51.7 reviews some of the potential pitfalls of the ALT and gives some suggestions.

Keywords

Accelerated life test (ALT) · Reliability test

Abstract

Accelerated life test (ALT) is a widely used method during product design with the aim to obtain reliability information on components and subsystems in a timely manner. Different types of ALTs provide different information about product and its failure mechanisms. To ensure that the ALTs can assess the product reliability accurately, quickly, and economically, designing efficient test plans is critically important. This chapter provides a limited discussion and description of the models and methods popularly used in ALT. The introduction describes the

51.1 Introduction

In reliability engineering, reliability tests are always used to assess the reliability indices and improve the design of products. To assess the reliability indices, it is mainly concerned with how to test or estimate them accurately, quickly, and economically. Determining a suitable statistical theory, method and technology for experiment design and data analysis is the key of conducting reliability tests, and it is often called statistical based reliability test (SRT). To improve the reliability of products, it mainly focuses on how the processes of design, material selection, manufacture, assemble, and application affect the storage, performance, and maintenance of the product. The key factor of achieving goals is the profound understanding of the performance evolution law of a particular product throughout the whole life cycle. This type of test has a higher requirement for engineering experience, and it is often called engineering based reliability test (ERT). Of course, this classification is only to emphasize the

Q. He (✉) · W.-H. Chen · J. Pan
 School of Mechanical Engineering, Zhejiang Sci-Tech University,
 Hangzhou, China
 e-mail: heqingchuan@zstu.edu.cn; chenwh@zstu.edu.cn;
 panjun@zstu.edu.cn

different focus of the two types of tests. In practice, to carry out a SRT correctly, the engineering elements, such as usage conditions, failure mode, failure mechanism, test equipment, and cost limits, should be specified. In an ERT, there is also a large quantity of data that should be gathered and analyzed based on statistics.

In practice, ERT is often used in conjunction with SRT, and they play different roles in various stages of whole life cycle of a product. According to the testing purposes, the reliability test could be further classified into reliability growth test (RGT), reliability qualification test (RQT), reliability screening test (RST), reliability acceptance test (RAT), and reliability determination test (RDT). According to the relationship between the test stress and the normal work stress, the reliability test could be divided into traditional test and accelerated test (AT). The types of the major reliability tests are shown in Fig. 51.1 [1].

In general, as shown in Fig. 51.1, the major implemented reliability tests are RGT, RQT, RST, RAT, and RDT corresponding to the design, finalization of design, production, delivery, and use phases of the product whole life cycle. The RGT and RST belong to ERT. The RQT, RAT, and RDT belong with SRT. The statistical inference methods used in RQT and RAT are mainly the hypothesis test. They are often called reliability verification test (RVT). The statistical inference methods used in RDT are mainly the parameter estimation.

In the period when the subject of reliability just formed, products usually have low reliability and short lifespan, and the reliability test can be carried out by simulating the

actual usage conditions. However, with the improvement of product reliability, this type of test is difficult to induce product failure effectively and cannot obtain adequate failure data within acceptable test time and cost. To solve this problem, the AT method is developed: The sample is tested within an environment more severe than the sample would experience during normal operating conditions. Data is collected from high stress levels and also is used to predict the product life under the normal stress level and to improve the product. Among the ATs, the ERT mainly includes the accelerated RGT, the highly accelerated life test (HALT) and the highly accelerated stress screening test (HASS). The HALT and HASS are often referred to as the reliability enhancement test (RET). The SRT mainly includes the accelerated life test (ALT) and the accelerated degradation test (ADT). ALT is a widely used method for rapidly obtaining reliability information and evaluating the useful life of product in a relatively short period. To ensure that the reliability of products can be assessed accurately, quickly, and economically, designing an efficient plan is a critical step before conducting the ALT, and that requires the support of relevant statistical theories. With the promotion of the national strategy of civil-military integration, the ALT will be widely applied in the research and development (R&D) of products, and the ALT plan design theory will also face more challenges. Based on the overview and prospect on designing ALT plan given in [1], this chapter provides a limited discussion and description of the models and methods popularly used in ALT.

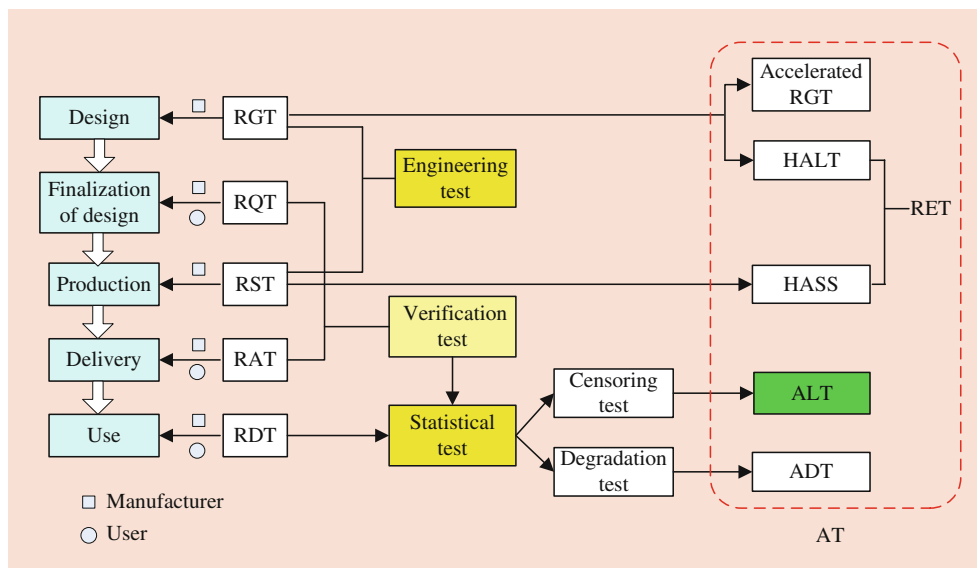


Fig. 51.1 Types, applications, and developments of reliability tests

51.2 Types of ALT

Different types of ALTs provide different information about product and its failure mechanisms. Generally, ALT experiments can be conducted using two approaches. The first is conducted by accelerating the use rate of the product at normal operating conditions such as switches and printers, and also home appliances such as refrigerator compressor and washing machines that are used only a fraction of time in one day. The manner in which the usage rate is increased depends on the product. For example, the median life of a switch for a light is 20 years (or 72,000 times), based on an assumed usage rate of ten times one day. If the switch is tested at 1200 h^{-1} , the median life is reduced to roughly 60 h. ALTs with increased usage rate attempt to simulate actual use. Therefore, the other environmental factors should be controlled to mimic actual use environments. If the usage rate is too high, it cannot have test items cool down between cycles of operation and thus can result in anomaly failure.

The second is conducted by subjecting a product to stresses severe-than-normal operating conditions to accelerate the failure. Depending on the nature of the product, life tests are accelerated by exposing the product to higher levels of accelerating variables such as temperature, pressure, and voltage. Thus, a sample at a high stress will generally fail more rapidly than it would have failed at low stress. With the premise that the failure modes and failure mechanisms are the same as those under normal stress, the samples are tested under stress levels higher than normal, and the lifetime of product under normal stress level can be estimated by extrapolating the life information of samples under high stress levels to normal level based on the life-stress relationship.

Different types of stress loadings can be considered when an ALT is conducted. Figure 51.2 shows four different types

of stress loading applied in ALTs, which are constant stress, step stress, progressive stress, and cyclic stress, respectively. The four loading schemes correspond to the constant stress ALT (CSALT), step stress ALT (SSALT), progressive stress ALT (PSALT), and cyclic stress ALT, respectively. The constant stress is time independent and also the most commonly used in engineering. The other three stress loading schemes is time dependent which means the stress level varies with time. Usually, time-dependent stress loadings can yield failures more quickly. In CSALT, samples are tested at constant, severer-than-normal operating conditions until either all samples fail or the test is terminated. In SSALT, the stress is increased at prespecified times or upon the occurrence of a fixed number of failures until either all the samples fail or the time period at the maximum stress level ends. In PSALT, the stress level is continuously increased as time goes on. In cyclic stress ALT, the stress level is periodically changed over time. The CSALT is applied to many products because it is easier to carry out, but need more samples and a long time at low stress levels to yield failure data. The SSALT, PSALT, and cyclic stress ALT could precipitate failures more rapidly. However, it is very difficult to model the life-stress relationship and, hence, to quantitatively predict the lifetime under normal usage conditions.

The ALT can be classified in other ways. Usually, two censoring schemes are widely used in ALT: time censoring (the type I censoring), where the number of actual failures is random upon the completion of test, and failure censoring (the type II censoring), where the total test duration is random at the end of test when certain number of failures is observed. These two kinds of censoring scheme correspond to the type I ALT and type II ALT. The two strategies of inspection performance of samples include continuous and periodic inspection, which can generate life data and group data, respectively. The number of accelerated stresses included single stress, two stresses, and multiple stresses. Usually,

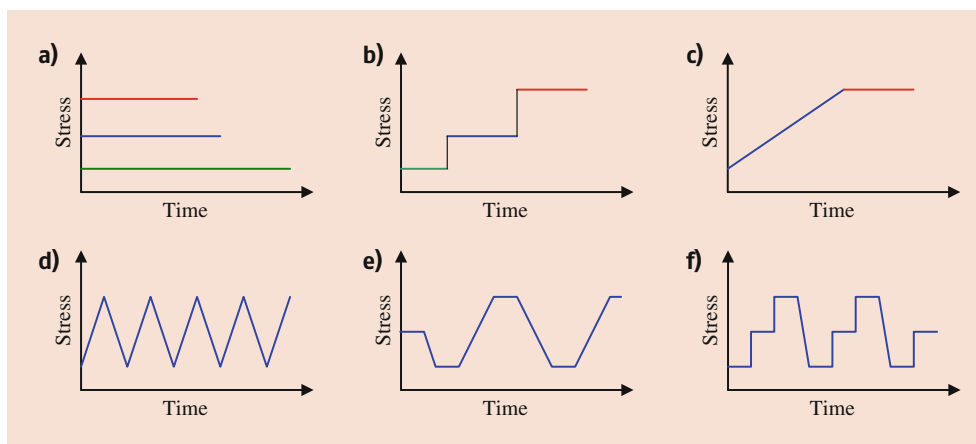


Fig. 51.2 Different types of accelerated stress loadings. (a) Constant stress. (b) Step stress. (c) Progressive stress. (d) Cyclic stress 1. (e) Cyclic stress 2. (f) Cyclic stress 3

the number of test stress is greater than or equal to three. They correspond to the single-stress ALT, two-stresses ALT, and multiple-stresses ALT, respectively. Comparing with the single-stresses ALT, the multiple-stresses ALT can simulate usage conditions better and can make the failure of products faster. With the rapid development of environment simulation technology, testing equipments used for multiple-stresses ALT are developed. All these equipments can load two or more environment stresses (such as the temperature and humidity, temperature and vibration, thermal and vacuum, and temperature and humidity and vibration) on products. Therefore, multiple-stresses ALTs begin to be used in engineering. For more details the reader is referred to Chap. ▶ 12 of Patrick and Andre [2].

51.3 Life Distribution

In reliability engineering, it is significant to determine which distribution best fits a set of data and to derive estimates of the distribution parameters. Let the symbol T denote lifetime for components, systems, etc. The cumulative distribution function (CDF) $F(t) = \Pr(T \leq t)$ is the probability of failure by time t . The probability density function (pdf) $f(t)$ is the derivative of the CDF. That is, $f(t) = dF(t)/dt$. Determining a probability distribution for lifetime means specifying either $F(t)$ or $f(t)$. The exponential, Weibull, and lognormal distributions are the most commonly used distributions in reliability engineering. For the other members of the family, the reader is referred to Chaps. ▶ 4 and ▶ 5 of Meeker and Escobar [3].

51.3.1 Exponential Distribution

The exponential distribution is a single-parameter distribution. If lifetime T has an exponential distribution its CDF and pdf are

$$F(t; \lambda) = 1 - \exp(-\lambda t),$$

$$f(t; \lambda) = \lambda \exp(-\lambda t), \quad t \geq 0$$

where λ is the constant hazard rate. The mean life or mean time to failure (MTTF) is $1/\lambda$.

The λ is a significant characteristic of the exponential distribution. It should be observed that λ has the same value and is independent of time. The exponential distribution is used for random failures. One salient feature is that the same proportion of parts fail relative to the number of parts remaining. Also, note that the exponential distribution is usually not appropriate because it assumes a constant λ .

Thus, the practitioner should be cautioned against using the exponential distribution, unless the underlying assumption of a constant λ can be justified.

51.3.2 Lognormal Distribution

Lifetime T is lognormally distributed if the logarithm of the T is normally distributed. When lifetime T has a lognormal distribution, its CDF and pdf are

$$F(t; \mu, \sigma) = \Phi_{\text{nor}}\left(\frac{\log(t) - \mu}{\sigma}\right),$$

$$f(t; \mu, \sigma) = \frac{1}{\sigma t} \phi_{\text{nor}}\left(\frac{\log(t) - \mu}{\sigma}\right), \quad t > 0$$

where Φ_{nor} and ϕ_{nor} are the standard normal CDF and pdf, respectively. In particular,

$$\phi_{\text{nor}(z)} = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right)$$

The parameters (μ, σ) are the mean and the standard deviation of $\log(T)$, respectively. The lognormal p quantile is $t_p = \exp[\mu + \Phi_{\text{nor}}^{-1}(p)\sigma]$. The lognormal distribution is used to model situations where large occurrences are concentrated at the tail (left) end of the range. It is based on the normal distribution, but is a more versatile distribution than the normal as it has a range of shapes, and therefore is often a better fit to reliability data, such as for products with wearout and fatigue characteristics.

51.3.3 Weibull Distribution

The Weibull distribution is developed in 1939 by Waloddi Weibull, who presented it in detail in 1951. If lifetime T has a Weibull distribution its CDF and pdf are

$$F(t; \eta, m) = 1 - \exp\left[-\left(\frac{t}{\eta}\right)^m\right],$$

$$f(t; \eta, m) = \frac{m}{\eta} \left(\frac{t}{\eta}\right)^{m-1} \exp\left[-\left(\frac{t}{\eta}\right)^m\right], \quad t > 0$$

where m is the shape parameter and η is the scale parameter. Note that η is approximately the 0.63 quantile of the Weibull distribution, or characteristic life. If, however, failures do not start at $t = 0$, but only after a finite time γ , then the CDF takes the form

$$F(t; \eta, m, \gamma) = 1 - \exp \left[- \left(\frac{t - \gamma}{\eta} \right)^m \right],$$

that is, a three-parameter distribution. The γ is called failure free time, location parameter, or minimum life.

The Weibull distribution is the most popular statistical distribution used in reliability engineering. It has the great advantage by adjusting the distribution parameters. It can be approximated to other distributions under special or limiting conditions. The Weibull distribution is widely used because a wide diversity of hazard rate curves can be modeled with it.

- If $m = 1$, the exponential reliability function (constant hazard rate) results with $\eta = \text{mean life } (1/\lambda)$ can be obtained. It is usually associated with useful life.
- If $m < 1$, a decreasing hazard rate reliability function can be obtained. It is usually associated with infant mortality, sometimes referred as early failures.
- If $m > 1$, an increasing hazard rate reliability function can be obtained. It is usually associated with wearout, corresponding to the end life of the product with closer inter-arrival failure times.
- When $m = 3.5$, for example, the distribution approximates to the normal distribution.
- If $m > 6$, it may reflect an accelerated rate of failures and fast wearout. Furthermore, it is time to doubt the accuracy of the analytical results.
- If $m > 10$, it is time to highly doubt the analytical results. Such a high β is fairly rare in practice.

The lognormal distribution and Weibull distribution are also called log-location-scale distributions. A random variable Y has a location-scale distribution if its CDF can be written as:

$$F(y; \mu, \sigma) = \text{Pr}(Y \leq y) = \Phi \left(\frac{y - \mu}{\sigma} \right),$$

where μ is a location parameter, σ is a scale parameter, and Φ does not depend on any unknown parameters. In many reliability applications, it is assumed that $\log(T)$ has a location-scale distribution. Then T is said to have a log-location-scale distribution.

51.4 Life-Stress Models

In reliability physics and engineering, the development and use of the acceleration factor (AF) and the life-stress relationship are vitally important to the theory of accelerated testing. The AF permits one to take time-to-failure data rapidly under accelerated test conditions, and then to extrapolate

the accelerated test results for a given set of operational conditions. The AF can be defined as given (51.1):

$$\text{AF} = \frac{t_{\text{normal}}}{t_{\text{acceleration}}} \quad (51.1)$$

where the t_{normal} and $t_{\text{acceleration}}$ are the time-to-failure data under normal operating conditions and accelerated stress conditions, respectively. Since t_{normal} may take many years to occur, then experimental determination of the AF is usually impractical. However, if one has proper life-stress relationship models then one can use these models to derive the AF.

A life-stress relationship can be one of the empirically derived relationship or a new one for the particular stress and application. It is applied to describe a life characteristic of the distribution from one stress level to another. The life characteristic can be any measure such as the mean, median, etc. For example, when considering the Weibull distribution, the scale parameter η is chosen to be the “life characteristic,” while m is assumed to remain constant across different stress levels. This assumption implies that the same failure mechanism is observed at different stresses. There are many life-stress models which have been used successfully in practice. Generally, the most commonly used life-stress relationship models include exponential model and power model.

51.4.1 Exponential Model

In this model, there is an exponential relationship between the life characteristic and the stress, which can be written as:

$$\text{Life} = A \times e^{-B \times \text{Stress}} \quad (51.2)$$

where A is an empirical constant and B can be a constant or function describing this relationship. Following are the most commonly used models involving stresses caused by temperature and humidity.

- Arrhenius Models

The Arrhenius equation is used to relate the rate of a chemical reaction R to temperature, which can be written as:

$$R(T) = \gamma_0 \exp \left(\frac{-E_a}{k_B \times T} \right) \quad (51.3)$$

where the γ_0 and the activation energy E_a are constants that depend on material properties, failure mechanism, and test methods; $k_B = 8.62 \times 10^{-5} \text{eV/K}$ is Boltzmann’s constant in units of electron volts per °C. The temperature $T = \text{temp } ^\circ\text{C} + 273.15$ is the temperature in Kelvin. Let T_{Field} and T_{test} be equal to absolute Kelvin temperatures at field

and test level, respectively. Then the Arrhenius acceleration factor can be written as:

$$AF = \exp \left\{ \frac{E_a}{k_B} \left(\frac{1}{T_{\text{Field}}} - \frac{1}{T_{\text{Test}}} \right) \right\}. \quad (51.4)$$

Based on (51.3), life is nonlinear in the single-stress variable temperature (T). Empirical observations suggested that the Arrhenius model has various applications, but it is most commonly used to estimate the acceleration factor for electronic component operating at a constant temperature. However, the nature of the failure mechanism may limit the range of temperature over which the Arrhenius relationship is adequate. The Arrhenius acceleration factor is also very sensitive to the value of E_a due to its exponential nature, therefore the accuracy of estimating E_a is very important. The E_a has a specific meaning as an atomic or material property. It simply becomes an empirical constant appropriate for use with a particular failure mechanism. There are no predetermined generic values of E_a for the specific failure mechanisms due to variation in parts characteristics. The only reliable way to obtain E_a is to conduct a series of acceleration tests.

- Eyring Model

The Eyring model is usually applied to combine the effect of more than one independent stress variables assuming no interactions between the stresses. A generic form of the Eyring equation can be written as:

$$\text{Life} = \exp \left\{ \frac{E_a}{k_B T} f_1(S_1) f_2(S_2) \right\} \quad (51.5)$$

where $f_1(S_1)$ and $f_2(S_2)$ are the factors for other applied stresses, such as temperature, humidity, voltage, current, vibration, and so on. A form of the Eyring model for the influence of voltage (V) in addition to temperature (T) is:

$$\text{Life} = AV^{-B} \exp \left\{ \frac{E_a}{k_B T} + \left(C + \frac{D}{T} \right) V \right\} \quad (51.6)$$

where A , B , C , and D are unknown parameters. Eyring also suggests the following generalized model, which considers the influences of temperature T and of further stresses S_i as:

$$AF = (T_2/T_1)^{-m} \exp \left\{ \frac{E_a}{k_B} \left(\frac{1}{T_{\text{Field}}} - \frac{1}{T_{\text{Test}}} \right) \right\} \exp \left\{ S_{\text{Field}} \left(C + \frac{D}{k_B T_{\text{Field}}} \right) - S_{\text{Test}} \left(C + \frac{D}{k_B T_{\text{Test}}} \right) \right\}. \quad (51.7)$$

See Meeker and Escobar [4] and the above references for more information on the Eyring model.

51.4.2 Power Model

A simple power relationship between the performance measure and the stress variable can be written as:

$$\text{Life} = A \times \text{Stress}^{-B} \quad (51.8)$$

where A is an empirical constant and B can be a constant or function describing this relationship. Following are the most commonly used power models involving stresses caused by temperature cycling, voltage, vibration, etc.

- Coffin-Manson Model

Models for mechanical failure, material fatigue, or material deformation are not forms of the Eyring model. These models typically have terms relating to cycles of stress or frequency of change in temperatures. The Coffin-Manson model is used successfully to model crack growth in solder and other metals due to repeated temperature cycling as equipment is turned on and off. It provides a relationship between life in thermal cycles, N_f , and plastic strain range $\Delta\gamma_p$, and says that the number of cycles to failure is

$$N_f(\Delta\gamma_p)^m = \text{Constant}, \quad (51.9)$$

where m is an empirical fatigue constant. During thermal cycling, the strain range caused by the mismatch of the coefficients of thermal expansion between solder and other materials is proportional to the cycling temperature excursion $\Delta T = T_{\text{Max}} - T_{\text{Min}}$. The acceleration factor can be approximated by:

$$AF = \left(\frac{\Delta T_{\text{Test}}}{\Delta T_{\text{Field}}} \right)^m \quad (51.10)$$

In the case of low cycle fatigue, the acceleration factor is typically used to the number of thermal cycles rather than the temperature exposure time. There are extensions of the Coffin-Manson model which account for the effect of temperature transition during thermal cycling.

- Voltage Acceleration Models

A simple model having only voltage V dependency takes the form:

$$AF = \left(\frac{V_{\text{Test}}}{V_{\text{Field}}} \right)^B \quad (51.11)$$

An alternative exponential voltage model takes the form of the following:

$$AF = \exp [B (V_{\text{Test}} - V_{\text{Field}})] \exp \left[\frac{E_a}{k_B} \left(\frac{1}{T_{\text{Field}}} - \frac{1}{T_{\text{Test}}} \right) \right] \quad (51.12)$$

where B is voltage acceleration parameter. Failure mechanisms of electronic components usually follow this relationship over most of their life for dependence on voltage.

- Vibration Acceleration Models

Most vibration models are based on the S-N curve. The relationship between peak stress σ and the number of cycles to failure N can be expressed as $N\sigma^b = \text{Constant}$ (high cycle fatigue). Assuming the linear relationship between the stress and acceleration G during vibration, the model takes the following form:

$$\text{Sinusoidal vibration : } AF = \left(\frac{G_{\text{Peak-Test}}}{G_{\text{Peak-Filed}}} \right)^b \quad (51.13)$$

$$\text{Random vibration : } AF = \left(\frac{G_{\text{Test}}}{G_{\text{Field}}} \right)^n \quad (51.14)$$

where b is the slope of the S-N line in the log-log scale and has different values for different materials. It is typical for sinusoidal vibration to measure life in vibration time or a number of cycles.

There are other models including the mix of exponential and power model or other physical laws, such as temperature-humidity model. Relative humidity is an environmental variable that can be combined with temperature to accelerate corrosion or other chemical reactions. As a special case of the Eyring model, Peck's equation is the most commonly used model to address the combined effect of temperature and humidity. The acceleration factor correlating product life in the field with test period can be expressed as:

$$AF = \left(\frac{RH_{\text{Test}}}{RH_{\text{Field}}} \right)^m \exp \left[\frac{E_a}{k_B} \left(\frac{1}{T_{\text{Field}}} - \frac{1}{T_{\text{Test}}} \right) \right] \quad (51.15)$$

where m is a constant, typically ranging between 2.0 and 4.0, and RH = relative humidity measured as percent. Usually, Peck's model can only be applied to wearout failure mechanisms, including electromigration, corrosion, dielectric breakdown, and dendritic growth. Another temperature-humidity model is Lawson model, which is based on the water absorption research:

$$AF = \exp \left[\frac{E_a}{k_B} \left(\frac{1}{T_{\text{Field}}} - \frac{1}{T_{\text{Test}}} \right) \right] \exp [b (RH_{\text{Test}}^2 - RH_{\text{Field}}^2)] \quad (51.16)$$

where b is an empirical humidity constant based on water absorption.

Example 51.1 An electrical insulator is rated for normal use at 12 V. Prior tests show this insulator can operate over 24 V and a life-stress exponent of the power model was found to be $B = 6$. How long must one run an ALT at 24 V to conduct an equivalent B_5 life if the B_5 life at 12 V is desired to be at least 20 years?

Let life $\text{Life} = A \times V^{-B}$ represent the life relationship, and applying (51.11) to calculate the acceleration factor due to voltage only:

$$AF = \left(\frac{24 \text{ V}}{12 \text{ V}} \right)^6 = 64$$

Thus

$$\begin{aligned} \text{Test time at 24 V} &= \frac{20 \text{ years}}{AF} = \frac{20 \text{ years} \times 8760 \text{ h/year}}{64} \\ &= 2737.5 \text{ h.} \end{aligned}$$

51.5 Statistical Analysis of ALT Data

An effective ALT conducted by using different types of stress loading, test stopping criteria, and performance inspection methods can produce different types of data such as failed and not failed (censored) data. Therefore, when dealing with the ALT data, appropriate data analysis methods must be used. Nelson and Meeker et al. are the pacesetters in discussing the method for statistical analysis of ALT data [2, 4]. Nelson and Meeker provide the basic analysis framework for ALT data. Although the ALT modes are extended, the statistical models are generalized and improved, and the framework has not yet been exceeded. In engineering, their researches and findings are widely used as important references in promoting the development of ALT design method. The most commonly used models and methods belong to the parametric statistics. The most widely used and advanced theory is the maximum likelihood estimation (MLE) theory for the location-scale distribution and the linear life-stress relationship. Nevertheless, the problems such as being difficult to determine the type of product life distribution and the lack of test data always bedevil the application of ALTs. See Meeker and Escobar [2] and Nelson [4, 6], and the above references for more information on statistical analysis of ALT data. There is also commercially available software designed to analyze ALT data. When sufficient data is available software packages such as ReliaSoft ALT can fit a statistical distribution to a life data set at each stress level and model the resulting life-stress relationship.

Generally, the first step in performing ALT data analysis is to determine an appropriate statistical distribution to de-

scribe lifetime at fixed levels of the accelerating variable(s). Typically, there is only one type of distribution used at all levels of stress. This implies that the failure mechanisms remain the same at different stress levels. The statistical model of CSALT includes the life distribution and the life-stress relationship. The statistical model of SSALT and PSALT includes the life distribution, the life-stress relationship, and also the equivalent principle of stress level transition. In studying the statistical theory of ALT, it began with supposing that the lifetime follows exponential distribution. From an engineering point of view, the ALT is primarily oriented to electronic products, and the exponential distribution is used widely as “standard distribution” in the reliability analysis. However, more engineers and applied statisticians find that it is more appropriate to describe the product life distribution as the function belong to the location-scale distribution (such as the Weibull distribution and lognormal distribution) than exponential distribution.

The process of finding the best statistical distribution is based on the failure data, goodness-of-fit tests, and engineering experience. Usually, probability plots are used to identify an appropriate distribution and to derive estimates of the distribution parameters. Probability plotting is a method for determining whether data conform to a hypothesized distribution. It in general involves a physical plot of the data on specially constructed probability plotting paper. The axes of probability plotting papers are transformed in such a way that the true CDF plots as a straight line. If the plotted data can be fitted by a straight line, the data fit the appropriate distribution. This approach can be easily implemented by hand under the given probability plotting paper. Nowadays computer software is used to assess the hypothesized distribution and determine the parameters of the underlying distribution by using MLE method or rank regression method. For example, probability plotting papers exist for all the major distribution,

such as normal, lognormal, Weibull, exponential, extreme value (see Reliasoft Weibull++ and Minitab 18).

Example 51.2 Assume that eight samples are tested. All of these items fail during the test after operating the following number of hours: 35, 40, 43, 47, 49, 54, 59, 63, 67, 74, 75, 77, 78, 86, 106, 108, 113, and 125. Find the best statistical distribution to describe lifetime of the product based on the observed failure data and determine the parameters.

Figure 51.3a, b show the Weibull distribution and lognormal distribution plotted with Minitab. In the two figures, the fitted line gives a good “eyeball fit” to the plotted data. It is very difficult to determine the best statistical distribution by using visual examination. However, the software employs more sophisticated mathematical methodology, and thus offers clear advantages by providing the capability to perform more accurate and versatile calculations and data plotting. See Fig. 51.3, the results (that is the value of AD and p) of the Anderson-Darling test show the lognormal distribution is a better one to describe lifetime of the product, and the location parameter μ is 4.213 and the scale parameter σ is 0.376.

The second step is to select (or create) a model to describe the relationship between the lifetime distributions and the accelerating variables. It is best to develop a model based on physical or chemical theory, empirical considerations, or both. Then, the pdf at the normal use stress level can be extrapolated based on the life-stress relationship and the characteristics of the distributions at each accelerated stress level. The life-stress relationships of single-stress, double-stresses, and multiple-stresses ALTs are single, binary, and multivariate functions, respectively. The number of stresses changes, and we must change the ALT plan design method with it. In practice, it is extremely difficult or impractical to

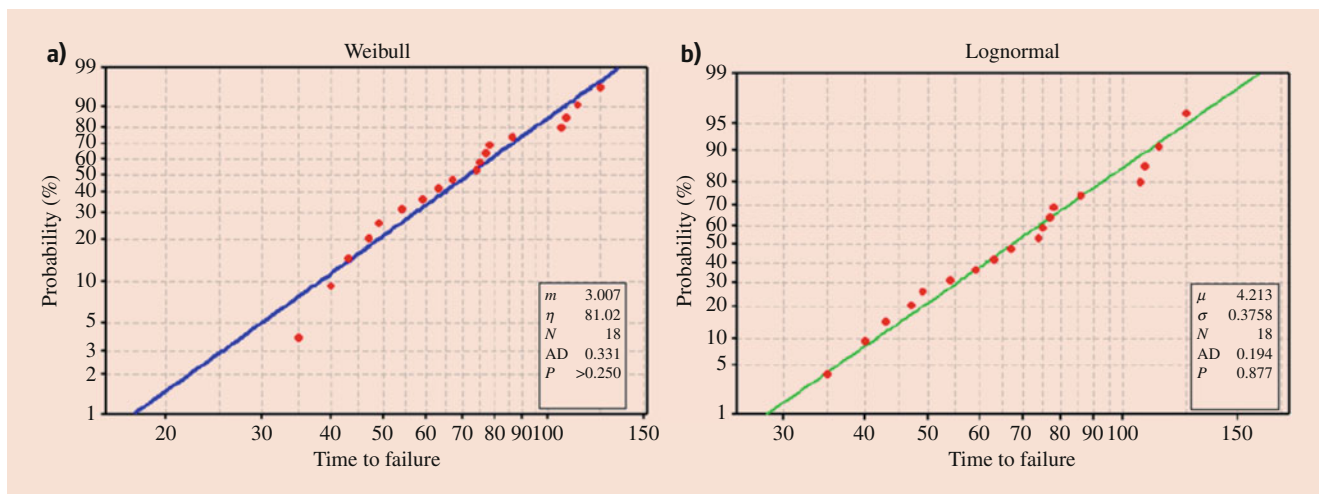


Fig. 51.3 Probability plotted with Minitab: (a) Weibull distribution plotted; (b) lognormal distribution plotted

verify acceleration relationships over the entire range of interest. Escobar and Meeker provide some basic guidelines for the use of acceleration models in reference [4]. Furthermore, for more details about time-to-failure modeling, the reader is referred to [6].

The next step is to assess statistical uncertainty in performing an ALT analysis. The statistical model and the statistical analysis method are the indispensable part for statistical analysis of ALT. The actual effectiveness of statistical inference is largely determined by whether the statistical model is suitable for engineering practices. Therefore, the statistical model should be tested before it is used in reliability assessment. The current theories provide methods for testing various life distributions. The verification on the life-stress relationship and the assumptions of cumulative damage is more concerned in engineering. The traditional theories on regression diagnosis are abundant, but few of them are applicable for censored data and non-normal distribution. In general, testing a model needs more sample sizes and stress levels than estimating model parameters. It is also difficult to validate the extrapolation effect of the model because of the long product life and high reliability. There needs to develop some new methods to deal with the uncertainty of the acceleration model. For more details about life data analysis process, the reader is referred to [1, 3, 5].

Example 51.3 To expedite product development of a capacitor, 45 capacitors are selected to conduct ALT. The capacitor is designed to operate at room temperatures up to 85 °C. The first group of 15 samples are tested at 1.4V₀ (V₀ is the

rated working voltage), second group at 1.6V₀, and third at 1.8V₀. The time-to-failure data obtained during this test are presented in Table 51.1.

The first step is to run life data analysis at each stress level and study how the accelerating variable affects the lifetime. For each level of the accelerating variable(s), plot the life data on probability paper for a suggested lifetime distribution. The distribution adequately describes the data if, for each individual condition, the points lie close to a straight line. ML is used to fit a line through the points at each level. A multiple probability plot shows fitted lines for all of the levels of the accelerating variable. This provides a convenient visual assessment of the assumption that all the samples have the same failure modes. If the assumption is reasonable, the lines will be approximately parallel.

Figure 51.4a, b show the Weibull and lognormal multiple probability plots, respectively. The points for each voltage stress level fall roughly along a straight line, and the lines appear to be reasonably parallel, but more parallel with the Weibull distribution. Thus, we confirm the consistency of the failure mechanisms at the three stress levels. It appears that both the Weibull and the lognormal distributions provide adequate descriptions of the ALT data. By comparison, the Weibull distribution is chosen as the life distribution. At each level of voltage stress, the Weibull distribution is fitted to the data, and estimates of η and m were computed by using the method of ML. Table 51.2 gives the ML estimates of η and m for each level of voltage stress obtained from Minitab.

Table 51.1 Accelerated test results for Example 51.3

Stress	Time to failure (h)
85 °C/1.4V ₀	89.3, 132.4, 276.3, 312.4, 346.6, 403.8, 476.8, 496.0, 682.4, 718.2, 897.4, 1008.3, 1036, 1094.5, 1136.0
85 °C/1.6V ₀	32.7, 34.8, 43.9, 57.2, 72.4, 99.5, 107.0, 123.7, 145.4, 180.1, 215.2, 235.5, 252.2, 280.0
85 °C/1.8V ₀	8.3, 9.6, 12.3, 17.5, 19.8, 24.0, 26.3, 36.2, 36.9, 45.5, 53.2, 63.6, 75.5, 76.3, 82.0

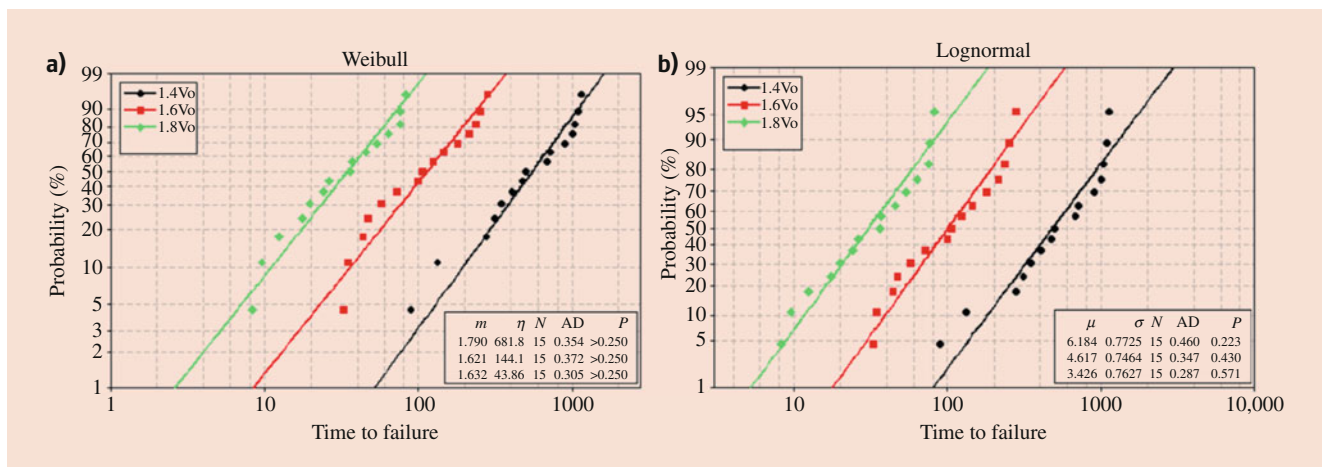


Fig. 51.4 Multiple probabilities plotted: (a) Weibull distribution plotted; (b) lognormal distribution plotted

The next step is to model a life-stress relationship. The inverse power relationship is widely used to relate the lifetime of a capacitor to voltage stress. Suppose that T (volt), the lifetime at volt, has a log-location-scale distribution with parameters (η, m) and that m does not depend on volt. Under the inverse power relationship, $\eta = \beta_0 + \beta_1 \ln(V)$. Fitting the model to data is accomplished by computing estimates for the parameters β_0 and β_1 . The result of fitting the inverse power relationship is

$$\hat{\eta}_0 = 1458 - 2571 \ln(V).$$

The estimate \hat{m} can be calculated by

$$\hat{m} = \frac{n_1 \hat{m}_1 + n_2 \hat{m}_2 + n_3 \hat{m}_3}{n_1 + n_2 + n_3} = 1.68$$

Then, the estimated rated lifetime at use conditions (85 °C/ V_0) can be estimated by

$$\hat{\eta} = 1458 - 2571 \ln(V_0) = 1458 \text{ h.}$$

And the reliability function at use conditions takes the form

$$R(t) = \exp \left[- \left(\frac{t}{\hat{\eta}_0} \right)^{\hat{m}} \right] = \exp \left[- \left(\frac{t}{1458} \right)^{1.68} \right].$$

Example 51.4 Consider the following times-to-failure data collected after testing 12 electronic devices at three different voltage stress levels and three temperature stress levels. On the i -th voltage stress level ($i = 1, 2, 3$) and the j -th temperature stress level ($j = 1, 2, 3$), the sample size n_{ij} is 8, the number of the failure data is r_{ij} , and the suspended time τ_{ij} is 10,000 h. Table 51.3 shows the time-to-failure data. As shown in Fig. 51.5, by using multiple probability plotting method, we found that the Weibull distribution is fitted to the

data at different stress levels. The estimates of η and m for each combination of voltage stress and temperature stress are obtained from Minitab. Suppose that the life-stress models are as follows:

$$\eta_{ij} = A \times V_i^{-C} \times \exp(B/T_j)$$

where the A , B , and C are unknown parameters. Find the reliability function at 120 V and 25 °C.

By using MLE, one can get the likelihood function as:

$$L = \prod_{i=1}^3 \prod_{j=1}^3 \frac{n_{ij}!}{(n_{ij} - r_{ij})!} \left\{ \prod_{k=1}^{r_{ij}} \frac{m t_{ij}^{m-1}}{\eta_{ij}^m} \exp \left[- \left(\frac{t_{(k)ij}}{\eta_{ij}} \right)^m \right] \right\} \times \exp \left[- \left(\frac{t_{(k)ij}}{\eta_{ij}} \right)^m \right]^{n_{ij} - r_{ij}}$$

Through maximizing $\ln L$, the maximum likelihood estimators of m , A , B , and C are the simultaneous solutions of four equations that are as follows:

$$\frac{\partial \ln L}{\partial m} = \sum_{i=1}^3 \sum_{j=1}^3 \left\{ \frac{r_{ij}}{m} + \sum_{k=1}^{r_{ij}} \ln t_{(k)ij} - r_{ij} \ln \eta_{ij} - \sum_{k=1}^{r_{ij}} \left(\frac{t_{(k)ij}}{\eta_{ij}} \right)^m \ln \frac{t_{(k)ij}}{\eta_{ij}} - (n_{ij} - r_{ij}) \left(\frac{\tau_{ij}}{\eta_{ij}} \right)^m \ln \frac{\tau_{ij}}{\eta_{ij}} \right\} = 0$$

$$\frac{\partial \ln L}{\partial A} = \sum_{i=1}^3 \sum_{j=1}^3 \frac{m}{A} \left\{ \sum_{k=1}^{r_{ij}} \left(\frac{t_{(k)ij}}{\eta_{ij}} \right)^m + (n_{ij} - r_{ij}) \left(\frac{\tau_{ij}}{\eta_{ij}} \right)^m - r_{ij} \right\} = 0$$

$$\frac{\partial \ln L}{\partial B} = \sum_{i=1}^3 \sum_{j=1}^3 \frac{m}{T_j} \left\{ \sum_{k=1}^{r_{ij}} \left(\frac{t_{(k)ij}}{\eta_{ij}} \right)^m + (n_{ij} - r_{ij}) \left(\frac{\tau_{ij}}{\eta_{ij}} \right)^m - r_{ij} \right\} = 0$$

Table 51.2 ML estimates for Example 51.3

V_i	1.4 V_0	1.6 V_0	1.8 V_0
$\hat{\eta}_i$	681.8	144.1	43.9
\hat{m}_i	1.79	1.62	1.63

Table 51.3 Time to failure for Example 51.4 (h)

Sample ID	Voltage (V)/Temperature (°C)								
	120/55	120/75	120/85	130/55	130/75	150/85	150/55	150/75	150/85
1	1523.8	138.4	255.1	1003.6	429.4	214.8	916.28	170.4	112.7
2	2916.7	873.9	492.0	2218.9	510.3	391.1	1036.5	237.3	203.1
3	3246.1	1221.6	640.0	2901.3	532.4	484.8	1748.5	413.1	245.2
4	3442.2	1316.7	682.5	3289.1	781.3	498.8	2959.4	917.1	462.7
5	6067.8	1557.0	685.0	4490.4	1337.8	573.0	3148.6	958.3	588.1
6	7169.1	1765.1	696.2	5220.8	1827.1	616.3	3841.5	1310.7	650.9
7	7705.3	2678.7	814.0	8320.5	2001.6	908.0	5221.9	1320.2	797.1
8	Suspended at 10,000	2914.8	1963.1	9043.8	2532.7	1643.4	6368.0	1448.1	1010.1

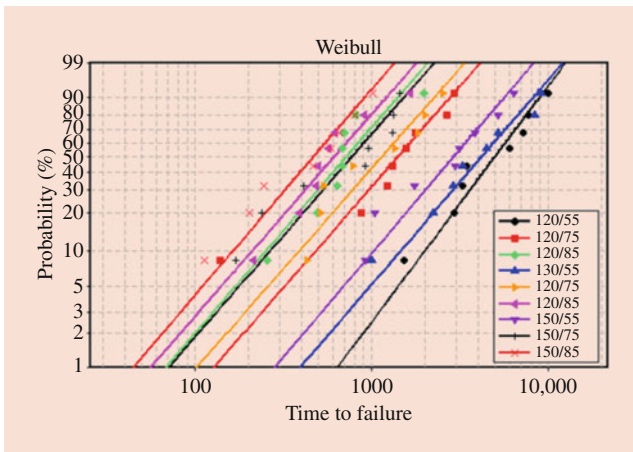


Fig. 51.5 Multiple Weibull distribution probability plotted

$$\frac{\partial \ln L}{\partial C} = \sum_{i=1}^3 \sum_{j=1}^3 m \ln V_i \left\{ r_{ij} - \sum_{k=1}^{r_{ij}} \left(\frac{t_{(k)ij}}{\eta_{ij}} \right)^m - (n_{ij} - r_{ij}) \left(\frac{\tau_{ij}}{\eta_{ij}} \right)^m \right\} = 0$$

Then, one can obtain that: $\hat{m} = 1.8$, $\hat{A} = 0.1$, $\hat{B} = 7543.3$ and $\hat{C} = 2.5$. The estimate lifetime at use conditions (120 V/25 °C) can be estimated by

$$\hat{\eta} = \hat{A} \times V^{-\hat{C}} \times \exp\left(\frac{\hat{B}}{T}\right) = 0.1 \times 120^{-2.5} \times \exp\left(\frac{7543.3}{273 + 25}\right) = 62418 \text{ h.}$$

And the reliability function at 120 V and 25 °C takes the form

$$R(t) = \exp\left[-\left(\frac{t}{\hat{\eta}}\right)^{\hat{m}}\right] = \exp\left[-\left(\frac{t}{62,418}\right)^{1.8}\right].$$

51.6 Design of ALT Plans

Finding an appropriate ALT method and an optimal test plan has become the common goal of all engineers. When designing an ALT plan, the type of stress loading, the number of stress levels and samples to be allocated to each stress level, censoring schemes, and also an applicable lifetime distribution and a life-stress models need to be first determined under many constraints, such as the limited test time, budget, and availability of resources. Designing optimal ALT plan should fit for the stress loading mode, ALT model, testing condition limitation, and design objective.

There are various optimization objectives and constraints such as V-optimization, D-optimization single objective,

multiple objectives, cost limits, and resource limits. By using V optimization, one can obtain the optimal plan by minimizing the asymptotic variance of the MLE of the p -th quantile of the product life distribution under normal stress. By using D optimization, one can obtain the optimal plan by maximizing the determinant of the Fisher information matrix of MLE. The most widely used researched ALT plan is the V-optimal continuous-inspection type I censoring CSALT plan with the location-scale distribution and linear life-stress relationship. For an overview of ALT design and planning, the reader is referred to [1, 7].

51.6.1 Method for Planning Single-Stress CSALTs

During planning ALTs, firstly suppose that [7]: (1) the logarithm life of product follows the Weibull distribution or lognormal distribution, and the CDF is $F(y; \mu, \sigma) = \Phi[(y - \mu)/\sigma]$, where μ is the location parameter, σ is the scale parameter, and $\Phi(\bullet)$ is the standard extreme value or the standard normal distribution. (2) The location parameter μ is a linear function of the standardized stress ξ ($0 \leq \xi \leq 1$), or the life-stress relationship is $\mu(\xi) = \gamma_0 + \gamma_1 \xi$, where $\gamma_1 < 0$. The Arrhenius model and the inverse power law model, which are the most widely used in engineering, both could be transformed into linear life-stress relationship. (3) The scale parameter σ is a constant and independent of ξ . (4) The failure time is statistically independent. (5) The type I censoring CSALT is considered, and the censoring time at each stress level is τ .

Assume that there are k stress levels, the sample size allocated to the i -th level ξ_i ($i = 1, 2, \dots, K$) is N_i , and also the lifetime of the j -th ($j = 1, 2, \dots, N_i$) samples on ξ_i is (t_{ij}, δ_{ij}) . If the sample fails, then $\delta_{ij} = 1$. If the sample is censored, then $\delta_{ij} = 0$ and $t_{ij} = \tau$. The log likelihood function of the MLE is

$$\ln L = \sum_{i=1}^K \sum_{j=1}^{N_i} \left\{ \delta_{ij} \ln f(t_{ij}; \mu(\xi_i), \sigma) + (1 - \delta_{ij}) \times \ln(1 - F(\tau; \mu(\xi_i), \sigma)) \right\} \tag{51.17}$$

where $f(t_{ij}; \mu(\xi_i), \sigma)$ is the pdf of the extreme value or normal distribution. The problem of designing optimal ALT plan could be also expressed as given in the following. Find the number of stress levels K^* , the stress level ξ_i^* , and the sample location ratio p_i^* that minimize the asymptotic variance of the MLE for the p -th quantile y_p of lifetime distribution under normal stress level under some constraint conditions. These conditions are mainly composed of the prior estimate values $\gamma_{0,e}$, $\gamma_{1,e}$, and σ_e of the model parameters γ_0 , γ_1 , and σ ,

respectively, and also the censoring time τ and the failure probability p under normal stress level.

However, the statistically optimal plan may not apply to practice. Nelson and Meeker et al. suggested a “compromise plan” with three or four stress levels can be used. The middle stress level can be used to test the life-stress relationship, prevent the test failures, and improve the plan robustness to the deviations of the statistical model and model parameters. Furthermore, Meeker also proposed the criterion that evaluates the ALT plan robustness to the deviations of the model parameters and product life distribution. The best plan of considering the estimation accuracy and robustness is the optimal compromise plan. This plan has three equally spaced test stresses and the sample location ratio of the middle stress level is 10% or 20%. The ratio at minimum stress level should be determined via optimization. This plan was also furtherly simplified by Meeker et al. to become the “4:2:1 plan,” which has three equally spaced levels and with the sample allocation ratio of 4:2:1 at the levels of lowest, the middle, and the highest stress.

51.6.2 Method for Planning Multiple-Stresses CSALTs

When the number of accelerated stresses is greater than 1, the life-stress relationship will turn into a binary or multivariate function. This leads to some problems, which are very different from planning single-stress ALTs. Escobar and Meeker carried out the earliest study on the theory and method of planning the optimal multiple constant stresses SALT for location-scale distribution. They used the assumptions given in Sect. 51.6.1, and generalized the life-stress relationship into a binary linear function $\mu(\xi_1, \xi_2) = \gamma_0 + \gamma_1\xi_1 + \gamma_2\xi_2$ (where $\gamma_i < 0$, $0 \leq \xi_i \leq 1$, and $i = 1, 2$), and then proved the following important conclusions. Firstly, the V-optimal MCSALT plan is not unique. Secondly, there is a type of V-optimal plan of which the stress level combinations $\xi_i^* = (\xi_{i1}^*, \xi_{i2}^*)$ ($i = 1, 2, \dots, K^*$) distribute on a straight line connecting the normal stress level (0, 0) and the highest stress level (1, 1). However, all parameters of the life-stress relationship cannot be determined by using these plans which are defined as optimal degenerated plan. Thirdly, each optimal degenerated plan corresponds to an infinite number of optimal nondegenerated plans. All parameters of the life-stress relationship can be determined by using the nondegenerated plans. The stress level combinations of optimal nondegenerated plans distribute on the life-stress relationship contour through the point $\xi_i^* = (\xi_{i1}^*, \xi_{i2}^*)$, and can be related with the stress level combinations of optimal degenerated plan by some equations.

Usually, the optimal plan is not just one. To get a determined plan one should restrict the arrangement mode of the

stress level combinations (called test points) in the feasible region of the test (called test region), and restrict the sample location ratio on test points. Escobar and Meeker proposed a method of obtaining the optimal nondegenerated plan (called splitting plan). Firstly, find the test point ξ_i^* and sample location ratios p_i^* by solving the optimization problem of single-stress ALT. And then find the two intersection points $\xi_{i,1}^*$ and $\xi_{i,2}^*$ of the life-stress relationship contour through the point ξ_i^* and the boundary of test region. The sample location ratio on $\xi_{i,1}^*$ and $\xi_{i,2}^*$ also should be inversely proportional to their distance to ξ_i^* . The splitting plan is the V-optimal plan, and is also the D-optimal plan among all V-optimal plans.

Furthermore, for a splitting plan, the number of test points and the difficulty of finding them increase with increasing in the number of stresses. In addition, the sample allocations at the test points are also reduced accordingly, and this will increase the risks of test failure. However, the degenerated plans are almost irrelevant to the dimension. The interaction effect between the stresses will also make the life-stress relationship to be a nonlinear function. In principle, with a little generalization, the splitting plan, orthogonal plan, and uniform plan are all applicable to the nonlinear life-stress relationship. However, a “chord method” for planning the V-optimal CSALT with time censoring and continuous inspection could be used to achieve a better plan [8]. For the problem of planning optimal MCSALT, whether the life-stress relationship is univariate or multivariate, linear or nonlinear, and whether the test region is rectangular or nonrectangular, the method could transform it into the problem of planning a single-stress ALT with linear life-stress relationship.

51.6.3 Design of SSALT Plans

Nelson firstly introduced the hypothesis of cumulative damage, which stated that the development of product damage under the same type of stress and failure mechanism was only related to the current state and current stress level, and was independent of the previous history of stress loading. Based on this hypothesis, Nelson established a rule of equivalent conversion between the life distributions and test times under different stress levels, and also proposed the theory and method of applying the step stress test to the SRT. In SSALT, the main method of estimating the model parameters is the MLE. The simple SSALT that has only two steps is generally used in practice. In addition, a three-step SSALT should be used to check whether the life-stress relationship is linear or not. The SSALT could save test costs although there is still no sufficient evidence to determine which is the best ALT mode among the CSALT and SSALT. When the total sample size and censoring time are the same, if only one test device is available, the SSALT can halved the test time by

comparing with CSALT. For more details about ALT design and planning, the reader is referred to [7, 9]. They give more detailed discussions of issues related to plan SSALT.

51.7 Some Pitfalls of Accelerated Testing

The aim of ALT is to obtain the reliability information of a product in a short period of time. There are numerous potential pitfalls in planning and conducting ALTs, as explained below.

In data analysis and interpretation stage, determining the appropriate accelerating variables and the associated life-stress relationship adequate for extrapolation are critical concerns. If the life-stress relationship does not reflect the actual failure processes, it is impossible to obtain inappropriate interpretations on the ALT data and assess the product reliability under the normal use conditions. Therefore, it is necessary to test the statistical model first before it is used in reliability assessment. The current theories provide methods for testing various common life distributions. However, from the aspect of engineering application, it is not easy to determine the life distribution and life-stress relationship of a specific product with competitive failure, because once there are more than one failure modes and failure causes, the data collection and the physical analysis on failure will become more difficult. The verification on the life-stress relationship and the assumptions of cumulative damage (for SSALT and PSALT) is more difficult. Testing a model needs more sample sizes and stress levels than estimating model parameters. For the ALT, it is difficult to directly validate the extrapolation effect of the model because of the long product life and high reliability. Furthermore, estimates and confidence intervals from ALTs also contain statistical uncertainty due to having only a limited amount of sample size and failure data. More extrapolation will exacerbate the problem. The Bayesian method and accelerated degradation test (ADT) can reduce the requirements on sample size to a certain extent, thus currently get more and more attention. Therefore, some new methods are needed to be dealt with the uncertainty of the acceleration model. Besides, for some products with complicated structures, their life distribution and life-stress relationship may be difficult to describe in a simple form.

When conducting ALT, the failure mechanisms at high levels of accelerating variables may not be the same as those at low levels. High levels of stresses may result in failure modes that are not representative of the field application or not occur in normal use of the product. Laboratory conditions are also not always similar to field conditions. There are more failures in the field than are predicted by analyzing the ALT data. During testing one only could do a periodic inspection that generates group data. Theoretically, there have special data analysis methods and optimal plan design for group data.

However, a systematic study on the accuracy of this method is not yet reported.

When planning ALT, different optimization criteria can be considered and different methods can be used for estimation of model parameters. However, each method has its inherent statistical properties and also, strictly speaking, no statistical model is exactly correct. Thus, the uncertainty reflected in the ALT plan need to be recognized. Furthermore, there is a lack of effective methods for planning ALTs of system-level products. For the system-level products, various problems, such as minimal sample size, competitive failure, complex stresses, and component relevance, often arise simultaneously, and there are some technical problems followed.

Note, the pitfalls during planning and conducting of ALTs must be considered. For more details about analysis of time-dependent accelerated testing, the reader is referred to [8, 10, 11]. In practice, pressure from management to reduce test time and cost can cause decisions that attempt to ignore these pitfalls. However, this will compromise the results of the ALT and result in misleading conclusions.

51.8 Summary

Reliability tests are always used to assess the reliability indices of product and improve products. ALT is a widely used method for rapidly obtaining reliability information and evaluating the useful life of the product in a relatively short period. In this chapter, we present a limited discussion and description of the models and methods popularly used in ALT, and also design of optimal test plans. Finally, we present some of the potential pitfalls the practitioner of ALT may face.

References

1. Chen, W.H., Gao, L., Pan, J., Qian, P., He, Q.C.: Design of accelerated life test plans—overview and prospect. *Chin. J. Mech. Eng.* **31**(2), 10–24 (2018)
2. Patrick, D.T., Andre, K.: *Practical Reliability Engineering*. Wiley, New York (2012)
3. Meeker, W.Q., Escobar, L.A.: *Statistical Methods for Reliability Data*. Wiley, New York (1998)
4. Escobar, L.A., Meeker, W.Q.: A review of accelerated test models. *Stat. Sci.* **21**(4), 552–577 (2006)
5. Nelson, W. (ed.): *Accelerated Testing: Statistical Models, Test Plans, Data Analyses*. Wiley, New York (2004)
6. McPherson, J.W. (ed.): *Reliability Physics and Engineering: Time-to-Failure Modeling*. Springer, Cham (2013)
7. Limon, S., Yadav, O.P., Liao, H.T.: A literature review on planning and analysis of accelerated testing for reliability assessment. *Qual. Reliab. Eng. Int.* **33**, 1–23 (2017)
8. Gao, L., Chen, W.H., Qian, P., Pan, J., He, Q.C.: Optimal time-censored constant-stress ALT plan based on chord of nonlinear stress-life relationship. *IEEE Trans. Reliab.* **65**(3), 1496–1508 (2016)

9. Elsayed, E.A.: Overview of reliability testing. *IEEE Trans. Reliab.* **61**(2), 282–291 (2012)
10. Meeker, W.Q., Escobar, L.A.: Pitfalls of accelerated testing. *IEEE Trans. Reliab.* **47**, 114–118 (1998)
11. Kirk, A.G., John, J.P.: *Next Generation HALT and HASS: Robust Design of Electronics and Systems*. Wiley, New York (2016)



Qingchuan He received his PhD from the Zhejiang Sci-Tech University in 2013. He is currently a lecturer at the National and Local Joint Engineering Research Center of Reliability Analysis and Testing for Mechanical and Electrical Products, Zhejiang Sci-Tech University, China. He works now on reliability testing and assessment.



Jun Pan received his PhD from the Zhejiang Sci-Tech University in 2010. He is currently a professor at the National and Local Joint Engineering Research Center of Reliability Analysis and Testing for Mechanical and Electrical Products, Zhejiang Sci-Tech University, China. He works now on reliability design, testing, and assessment.



Wen-Hua Chen received his Ph.D. degree from Zhejiang University, China, in 1997. He is currently a professor at the Zhejiang University and Zhejiang Sci-Tech University, and also the director of the National and Local Joint Engineering Research Center of Reliability Analysis and Testing for Mechanical and Electrical Products. He works now on reliability design, testing, and statistical analysis.



Accelerated Life Testing Data Analyses for One-Shot Devices

52

Narayanaswamy Balakrishnan and Man Ho Ling

Contents

52.1	Introduction	1039
52.2	Literature Reviews	1040
52.3	One-Shot Device Testing Data Under CSALTs and Models	1040
52.3.1	Exponential Models.....	1041
52.3.2	Weibull Models.....	1041
52.3.3	Gamma Models.....	1042
52.3.4	Generalized Gamma Models.....	1042
52.3.5	Proportional Hazards Models.....	1043
52.4	Likelihood Inference	1043
52.4.1	EM Algorithm.....	1043
52.4.2	Confidence Intervals.....	1045
52.4.3	Model Mis-Specification Analysis.....	1046
52.4.4	Model Selection.....	1047
52.4.5	Test of Proportional Hazard Rates.....	1050
52.4.6	Designs of CSALTs.....	1050
52.5	Minimum Density Power Divergence Estimation . . .	1050
52.5.1	Minimum Density Power Divergence Estimators	1050
52.5.2	Wald-Type Tests.....	1051
52.6	One-Shot Devices with Multiple Failure Modes	1052
52.6.1	Competing Risks Models for Independent Failure Models.....	1052
52.6.2	Copula Models for Correlated Failure Models.....	1053
52.7	Step-Stress Accelerated Life Tests	1054
52.7.1	Exponential Models.....	1054
52.7.2	Weibull Models.....	1055
52.7.3	Designs of Simple SSALTs.....	1055
References		1056

Abstract

One-shot device testing data arises from devices that can be used only once, for example, fire extinguishers, electro-explosive devices, and airbags in cars. In life tests, only the conditions of the tested devices at a specified time can be observed, instead of their actual lifetimes. Such data is therefore of either left- or right-censored. For these heavily censored data, there is an increasing need to develop innovative techniques for reliability analysis. In this chapter, we provide an overview of analyses of one-shot device testing data collected from accelerated life tests and discuss some statistical issues on the statistical estimation and inference as well as optimal designs of accelerated life tests for one-shot devices.

Keywords

One-shot device · Accelerated life tests · Constant-stress · Step-stress · Likelihood inference · Minimum density power divergence · Model selection · Model mis-specification · Multiple failure modes · Test plans

52.1 Introduction

One-shot device testing data analyses have recently received great attention in reliability studies. This chapter aims at providing the latest progress of statistical models and methods for analyzing one-shot device testing data collected from accelerated life tests. One-shot device testing data differs from typical data obtained by measuring lifetimes in standard life tests and poses a unique challenge in reliability analysis, due to a lack of lifetime information being collected from life tests. A one-shot device is a unit that is accompanied by

N. Balakrishnan (✉)
Department of Mathematics and Statistics, McMaster University,
Hamilton, ON, Canada
e-mail: bala@mcmaster.ca

M. H. Ling
The Education University of Hong Kong, Hong Kong, China
e-mail: amhling@eduhk.hk

an irreversible chemical reaction or physical destruction and therefore no longer functions properly after use. As a result, the actual lifetime of a one-shot device cannot be obtained from life tests, and thus one-shot device testing data is binary; researchers can only observe either a success or failure at the inspection time as each device can be tested only once. The lifetime is either left- or right-censored. If successful tests occur, it implies that the lifetimes are beyond their inspection times, leading to right-censoring. On the contrary, the lifetimes are before their inspection times, leading to left-censoring, if failed tests appear. Examples of one-shot devices include explosive devices in stockpiles of military weapons, missiles, rockets, fire extinguishers, and vehicle airbags [3, 20, 48]. Shaked and Singpurwalla [39] proposed a Bayesian approach to assess submarine pressure hull damage. Hwang and Ke [22] developed an iterative procedure to evaluate the storage life and reliability of one-shot devices. Subsequently, Pan and Chu [37] studied one-shot devices in a series system.

One-shot devices are required to perform its function only once. Therefore, they are normally highly reliable and with longer lifetimes under normal operating conditions; however, it is very unlikely to observe many failures under normal operating conditions in a short time. To accurately estimate the reliability with a few or even no failures becomes very challenging. Recently, accelerated life tests (ALTs) are often applied to address this practical issue. In ALTs, devices are subject to stress levels higher than normal operating conditions to induce early failures. Consequently, more failures can be obtained in a short time. As there is no interest in estimating the reliability under the higher-than-normal stress levels, ALT models are then typically used to extrapolate the data obtained from the elevated stress levels to estimate the reliability under normal operating conditions. Since, ALTs are efficient in capturing valuable lifetime information within a limited time and therefore become a substantial element in a wide variety of reliability studies for many decades, including batteries [32], tantalum capacitors [42], and power modules [33]. One may refer to recent books with chapters on ALT models [5, 31, 36, 38] for details. There are many types of ALTs. Constant-stress ALTs (CSALTs) wherein each device is subject to only one pre-specified stress level are frequently used. On the other hand, step-stress ALTs (SSALTs) apply stress to devices in the way that stress levels will increase at pre-specified times step-by-step. These two types of ALTs have received great attention in literature.

The rest of this chapter is organized as follows. Section 52.2 presents literature reviews of one-shot device testing data analyses. Section 52.3 describes one-shot device testing data collected from CSALTs and statistical models that have been studied for one-shot device testing data. Section 52.4 considers the likelihood estimation and inference for one-shot device testing and discusses some statistical issues, including model mis-specification, model selection,

testing for proportional hazards rates, and designs of CSALTs for one-shot devices. Sections 52.5–52.7 discuss some latest developments of one-shot device testing data analyses. A robust estimator and hypothesis tests for one-shot device testing data, based on minimum density power divergence measures, are presented in Sect. 52.5. Sections 52.6 and 52.7 consider analyses for one-shot device testing data with multiple failure modes and one-shot device testing data collected from SSALTs, respectively.

52.2 Literature Reviews

This section provides an overview of one-shot device testing data analyses under ALTs. Under CSALTs, Fan et al. [19] compared three different priors in the Bayesian approach for making predictions on the reliability at a mission time and the mean lifetime of electro-explosive devices under normal operating conditions. Balakrishnan and Ling [6–9] subsequently developed expectation-maximization (EM) algorithms for likelihood estimation based on one-shot device testing data under exponential, Weibull, and gamma distributions. In addition, several confidence intervals for the mean lifetime and the reliability at a mission time under normal operating conditions have been studied. Balakrishnan et al. [14, 15] recently developed weighted minimum density power divergence estimators for one-shot device testing data under exponential and gamma distributions. Ling et al. [25] considered proportional hazards models for analyzing one-shot device testing data. Apart from the statistical estimation and inference, Balakrishnan and Ling [10] developed a procedure to obtain CSALT plans with budget constraints for one-shot devices. Ling and Balakrishnan [24] also studied model mis-specification effects on one-shot device testing data analyses between Weibull and gamma distributions and Balakrishnan and Chimitova [4] conducted comprehensive simulation studies to compare several goodness-of-fit tests for one-shot device testing data. In addition, Balakrishnan et al. [11–13] extended this line of work by introducing competing risks models to analyze one-shot device testing data with multiple failure modes. On the other hand, Ling [23] recently presented an EM algorithm for one-shot device testing data collected from SSALTs and also studied SSALT plans for one-shot devices with lifetimes following exponential distributions.

52.3 One-Shot Device Testing Data Under CSALTs and Models

Suppose that CSALTs with I test groups are considered. For $i = 1, 2, \dots, I$, in the i -th test group, K_i one-shot devices are subject to J types of accelerating factors (i.e., temperature, humidity, voltage, and pressure) at stress levels

Table 52.1 One-shot device testing data with multiple accelerating factors, various stress levels, and different inspection times

Test group	Stress levels	Inspection time	# of tested devices	# of failures
1	$(x_{11}, x_{12}, \dots, x_{1J})$	τ_1	K_1	n_1
2	$(x_{21}, x_{22}, \dots, x_{2J})$	τ_2	K_2	n_2
\vdots	\vdots	\vdots	\vdots	\vdots
I	$(x_{I1}, x_{I2}, \dots, x_{IJ})$	τ_I	K_I	n_I

$\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$ and are inspected at time τ_i . The corresponding number of failures, n_i , is then recorded. The data thus observed can be summarized below in Table 52.1.

For notational convenience, we denote $\mathbf{z} = \{\tau_i, K_i, n_i, \mathbf{x}_i, i = 1, 2, \dots, I\}$ for the observed data and θ for the model parameters. Suppose the lifetime in the i -th test group has a distribution with the cumulative distribution function (cdf), $F(t; \Psi_i)$ with $\Psi_i = \Psi(\mathbf{x}_i; \theta)$. The observed likelihood function is then given by

$$L(\theta; \mathbf{z}) = C \prod_{i=1}^I [F(\tau_i; \Psi_i)]^{n_i} [R(\tau_i; \Psi_i)]^{K_i - n_i},$$

where $R(\tau_i; \Psi_i) = 1 - F(\tau_i; \Psi_i)$ and C is the normalizing constant. The data is used to determine some life characteristics of the devices under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, namely, the mean lifetime, μ , and the reliability at a mission time t , $R(t)$.

52.3.1 Exponential Models

Fan et al. [19], Balakrishnan and Ling [6, 7] studied one-shot device testing data under exponential distributions with single and multiple accelerating factors. Let E denote the lifetime that follows the exponential distribution. In the i -th test group, the cdf and the probability density function (pdf) of the exponential distributions with the rate parameter $\lambda_i > 0$ are given by

$$F_E(t; \lambda_i) = 1 - \exp(-\lambda_i t), \quad t > 0,$$

and

$$f_E(t; \lambda_i) = \frac{\partial F_E(t; \lambda_i)}{\partial t} = \lambda_i \exp(-\lambda_i t), \quad t > 0,$$

respectively. Its mean lifetime and reliability at a mission time t under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, are then given by

$$\mu_E = \int_0^\infty t f_E(t; \lambda_0) dt = \lambda_0^{-1},$$

and

$$R_E(t) = 1 - F_E(t; \lambda_0) = \exp(-\lambda_0 t),$$

respectively. As we need to extrapolate the data from the elevated stress levels, $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$, to the life characteristics of the devices under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, the parameter λ_i is related to the stress levels in a log-linear link function, with $x_{i0} = 1$, i.e.,

$$\ln(\lambda_i) = \sum_{j=0}^J e_j x_{ij}.$$

It is noting that many well-known stress-rate models, such as Arrhenius, inverse power law, and Eyring models, are all special cases of the log-linear link function, with appropriate transformations on the stress levels [44]. In this setting, the model parameters are $\theta_E = \{e_j, j = 0, 1, \dots, J\}$ to be estimated.

52.3.2 Weibull Models

As the exponential distributions enable us to model constant hazard rate only, Balakrishnan and Ling [8] further extended the work to Weibull distributions with multiple accelerating factors that vary both scale and shape parameters, because the Weibull distributions can model increasing, constant, and decreasing hazard rates and include the exponential distributions as a special case.

Let W denote the lifetime that follows the Weibull distribution. In the i -th test group, the cdf and the pdf of the Weibull distributions with the shape parameter $\eta_i > 0$ and the scale parameter $\beta_i > 0$ are given by

$$F_W(t; \eta_i, \beta_i) = 1 - \exp\left(-\left(\frac{t}{\beta_i}\right)^{\eta_i}\right), \quad t > 0,$$

and

$$\begin{aligned} f_W(t; \eta_i, \beta_i) &= \frac{\partial F_W(t; \eta_i, \beta_i)}{\partial t} \\ &= \frac{\eta_i}{\beta_i} \left(\frac{t}{\beta_i}\right)^{\eta_i - 1} \exp\left(-\left(\frac{t}{\beta_i}\right)^{\eta_i}\right), \quad t > 0, \end{aligned}$$

respectively. Its mean lifetime and reliability at a mission time t under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, are, respectively, given by

$$\mu_W = \int_0^\infty t f_W(t; \eta_0, \beta_0) dt = \beta_0 \Gamma\left(1 + \frac{1}{\eta_0}\right),$$

and

$$R_W(t) = 1 - F_W(t; \eta_0, \beta_0) = \exp\left(-\left(\frac{t}{\beta_0}\right)^{\eta_0}\right),$$

where $\Gamma(z) = \int_0^\infty x^{z-1} \exp(-x) dx$ is the gamma function. To extrapolate the data from the elevated stress levels, $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$, to the life characteristics of the devices under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, the parameters η_i and β_i are related to the stress levels in log-linear link functions, with $x_{i0} = 1$, i.e.,

$$\ln(\eta_i) = \sum_{j=0}^J r_j x_{ij}, \quad \text{and} \quad \ln(\beta_i) = \sum_{j=0}^J s_j x_{ij}.$$

Therefore, the model parameters are $\theta_W = \{r_j, s_j, j = 0, 1, \dots, J\}$ to be estimated.

52.3.3 Gamma Models

In addition to the Weibull distributions, gamma distributions can model increasing, constant, and decreasing hazard rates and include the exponential distributions as a special case. Balakrishnan and Ling [9] studied one-shot device testing data under gamma distributions with multiple accelerating factors that vary both scale and shape parameters.

Let G denote the lifetime that follows the gamma distribution. In the i -th test group, the cdf and the pdf of the gamma distributions with the shape parameter $\alpha_i > 0$ and the scale parameter $\beta_i > 0$ are, respectively, given by

$$F_G(t; \alpha_i, \beta_i) = \gamma\left(\alpha_i, \frac{t}{\beta_i}\right), \quad t > 0,$$

and

$$f_G(t; \alpha_i, \beta_i) = \frac{\partial F_G(t; \alpha_i, \beta_i)}{\partial t} = \frac{t^{\alpha_i-1}}{\Gamma(\alpha_i)\beta_i^{\alpha_i}} \exp\left(-\frac{t}{\beta_i}\right), \quad t > 0,$$

where $\gamma(z, s) = \int_0^s x^{z-1} \exp(-x) dx / \Gamma(z)$ is the lower incomplete gamma ratio. Its mean lifetime and reliability at a mission time t under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, are, respectively, given by

$$\mu_G = \int_0^\infty t f_G(t; \alpha_0, \beta_0) dt = \alpha_0 \beta_0,$$

and

$$R_G(t) = 1 - F_G(t; \alpha_0, \beta_0) = \Gamma\left(\alpha_0, \frac{t}{\beta_0}\right),$$

where $\Gamma(z, s) = 1 - \gamma(z, s)$ is the upper incomplete gamma ratio. Again, to extrapolate the data from the elevated stress levels, $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$, to the life characteristics of the devices under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, the parameters α_i and β_i are related to the stress levels in log-linear link functions, with $x_{i0} = 1$, i.e.,

$$\ln(\alpha_i) = \sum_{j=0}^J a_j x_{ij}, \quad \text{and} \quad \ln(\beta_i) = \sum_{j=0}^J b_j x_{ij}.$$

Therefore, the model parameters are $\theta_G = \{a_j, b_j, j = 0, 1, \dots, J\}$ to be estimated.

52.3.4 Generalized Gamma Models

Generalized gamma distribution that was introduced by Stacy [41] includes the preceding popular lifetime distributions. In this regard, we can consider this highly flexible distribution to analyze one-shot device testing data and used it to discriminate between the Weibull and the gamma distributions.

Let GG denote the lifetime that follows the generalized gamma distribution. In the i -th test group, the cdf and the pdf of the generalized gamma distributions with the shape parameters $\alpha_i > 0$, $\eta_i > 0$ and the scale parameter $\beta_i > 0$ are, respectively, given by

$$F_{GG}(t; \alpha_i, \eta_i, \beta_i) = \gamma\left(\alpha_i, \left(\frac{t}{\beta_i}\right)^{\eta_i}\right), \quad t > 0,$$

and

$$f_{GG}(t; \alpha_i, \eta_i, \beta_i) = \frac{\partial F_{GG}(t; \alpha_i, \eta_i, \beta_i)}{\partial t} = \frac{\eta_i t^{\eta_i \alpha_i - 1}}{\Gamma(\alpha_i) \beta_i^{\eta_i \alpha_i}} \exp\left(-\left(\frac{t}{\beta_i}\right)^{\eta_i}\right), \quad t > 0.$$

It is noting that the generalized gamma distribution becomes the Weibull distribution when $\alpha_i = 1$, while it becomes the gamma distribution when $\eta_i = 1$ and further reduces to the exponential distribution when $\alpha_i = \eta_i = 1$. Besides, its mean lifetime and reliability at a mission time t under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, are, respectively, given by

$$\mu_{GG} = \int_0^\infty t f_{GG}(t; \alpha_0, \eta_0, \beta_0) dt = \alpha_0 \Gamma\left(\beta_0 + \frac{1}{\eta_0}\right) / \Gamma(\beta_0),$$

and

$$R_{GG}(t) = 1 - F_{GG}(t; \alpha_0, \eta_0, \beta_0) = \Gamma \left(\alpha_0, \left(\frac{t}{\beta_0} \right)^{\eta_0} \right).$$

Once again, to extrapolate the data from the elevated stress levels, $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iJ})$, to the life characteristics of the devices under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, the parameters α_i , η_i and β_i are related to the stress levels in log-linear link functions, with $x_{i0} = 1$, i.e.,

$$\ln(\alpha_i) = \sum_{j=0}^J u_j x_{ij}, \quad \ln(\eta_i) = \sum_{j=0}^J v_j x_{ij}, \quad \text{and} \\ \ln(\beta_i) = \sum_{j=0}^J w_j x_{ij}.$$

Therefore, the model parameters are $\boldsymbol{\theta}_{GG} = \{u_j, v_j, w_j, j = 0, 1, \dots, J\}$ to be estimated.

52.3.5 Proportional Hazards Models

The former models are fully parametric models for one-shot device testing data. Ling et al. [25] proposed proportional hazards models that allow the hazard rate to change in a nonparametric way and require a link function to relate the stress levels to the lifetimes of the devices.

Under the proportional hazards assumption [16, 17], in the i -th test group, the cumulative hazard function at the inspection time τ_i is given by

$$H(\tau_i; v_i) = H_0(\tau_i) v_i,$$

where $H_0(\tau_i)$ is the baseline cumulative hazard function. This model is composed of two s -independent components. One component measures the influence of the stress factors, and the other measures the changes in the baseline. We now assume a log-linear link function to relate the stress levels to the lifetimes of the devices to obtain the cumulative hazard function as

$$H(\tau_i; v_i) = H_0(\tau_i) \exp \left(\sum_{j=1}^J h_j x_{ij} \right);$$

the corresponding reliability function is then

$$R_{PH}(\tau_i; v_i) = \exp(-H(\tau_i; v_i)) = \xi_i^{v_i} = \xi_i^{\exp(\sum_{j=1}^J h_j x_{ij})},$$

where $\xi_i = \exp(-H_0(\tau_i))$ is the baseline reliability at time τ_i and $x_{i0} = 1$. It is of interest to note that the baseline reliability

is decreasing in time and bounded between 0 and 1, that is, $0 < \xi_i < \xi_j < 1$ if $\tau_i > \tau_j$. Its reliability at a mission time τ_i under normal operating conditions, $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, is then given by

$$R_{PH}(\tau_i) = R_{PH}(\tau_i; v_0) = \xi_i^{v_0}.$$

Therefore, the model parameters are $\boldsymbol{\theta}_{PH} = \{\xi_i, i = 1, 2, \dots, I, h_j, j = 1, 2, \dots, J\}$ to be estimated.

We now present a connection between the proportional hazards model and the Weibull distribution having proportional hazards rates. In CSALTs, if the lifetimes of devices subject to elevated stress levels \mathbf{x}_i follow the Weibull distribution with the scale parameter $\beta_i = \exp(\sum_{j=0}^J s_j x_{ij})$ and the same shape parameter, $\eta_0 = \eta_1 = \dots = \eta_I = \eta$, the reliability of the Weibull distribution is

$$R_W(t, \eta_i, \beta) = \exp \left(- \left(\frac{t}{\beta_i} \right)^\eta \right) \\ = \exp \left(- \left(t^\eta \exp \left(- \sum_{j=0}^J \eta s_j x_{ij} \right) \right) \right), \quad t > 0.$$

The Weibull distribution with the constant parameter is satisfied with the proportion hazards assumption, and then the baseline reliability and the coefficients of stress factors are given by

$$\xi_i = \exp(-\tau_i^\eta \exp(-\eta s_0)), \quad i = 1, 2, \dots, I,$$

and

$$h_j = -\eta s_j, \quad j = 1, 2, \dots, J.$$

52.4 Likelihood Inference

Likelihood inference on one-shot device testing data under several popular distributions in reliability studies has extensively been studied [6–9]. This section provides EM frameworks to find the maximum likelihood estimates (MLEs) of parameters of interest. Besides, several confidence intervals for the corresponding parameters are presented in this section. Some statistical issues on model mis-specification and model selection are also discussed. CSALT plans for one-shot devices are finally considered.

52.4.1 EM Algorithm

In one-shot device testing, no actual lifetime can be observed. Due to these highly censored data, EM algorithms were developed for estimating the model parameters under the

forementioned distributions. The EM algorithm [30] is an efficient and powerful technique for finding the MLEs of the model parameters in case of the presence of missing data. Thus, the EM technique is extensively employed in various fields. The EM algorithm simply involves two steps in each iteration: (1) expectation (E)-step: approximating the missing data and (2) maximization (M)-step: maximizing the corresponding likelihood function.

Suppose the lifetime follows a distribution with the pdf $f(t; \theta)$. In the EM algorithm, we first consider the log-likelihood function based on the complete data as follows:

$$\ell_c(\theta) = \sum_{i=1}^I \sum_{k=1}^{K_i} \ln(f(t_{ik}; \theta)).$$

In the m -th iteration, the objective then is to update the estimate θ that maximizes the function

$$Q(\theta, \theta^{(m)}) = E_{\theta^{(m)}}[\ell_c(\theta)|\mathbf{z}]$$

based on the current estimate $\theta^{(m)}$. The current estimate $\theta^{(m)}$ is replaced with the updated estimate $\theta^{(m+1)}$ in the M-step, and then the conditional expectation $Q(\theta, \theta^{(m+1)})$ can be obtained based on the updated estimate $\theta^{(m+1)}$ in the E-step. These two steps are repeated until convergence is achieved to a desired level of accuracy. It can be seen that we actually solve the incomplete data problem by first approximating the missing data and then using the approximated values to find the estimate of the parameter vector as the solution for the complete data problem.

However, for the maximization problem, a closed-form solution cannot be found. The EM algorithm based on the one-step Newton-Raphson method needs to be employed for this purpose. It requires the second-order derivatives of the log-likelihood function with respect to the model parameters. For the present situation, let us define

$$\mathbf{I} = \left[\frac{\partial Q(\theta, \theta^{(m)})}{\partial \theta} \right]_{\theta=\theta^{(m)}}$$

and

$$\mathbf{J} = - \left[\frac{\partial^2 Q(\theta, \theta^{(m)})}{\partial \theta \partial \theta'} \right]_{\theta=\theta^{(m)}}$$

Then, we have the updated estimates as

$$\theta^{(m+1)} = \mathbf{J}^{-1} \mathbf{I} + \theta^{(m)}.$$

In many cases, the primary interest may be on some lifetime characteristics, such as the mean lifetime and the reliability under normal operating conditions, which are functions

of the model parameters. Hence, one can plug in the MLEs of the model parameters into the corresponding functions and obtain readily the MLEs of the parameters of interest.

Let's look at the example of exponential distributions. The log-likelihood function based on the complete data is given by

$$\ell_c = \sum_{i=1}^I \left(K_i \sum_{j=0}^J e_j x_{ij} \right) - \sum_{i=1}^I \left(\left(\sum_{k=1}^{K_i} t_{ik} \right) \exp \left(\sum_{j=0}^J e_j x_{ij} \right) \right),$$

and the conditional expectation is

$$Q(\theta_E, \theta_E^{(m)}) = \sum_{i=1}^I \left(K_i \sum_{j=0}^J e_j x_{ij} \right) - \sum_{i=1}^I \left(K_i t_i^* \exp \left(\sum_{j=0}^J e_j x_{ij} \right) \right).$$

The first and second derivatives with respect to the model parameters to maximize the quantity $Q(\theta_E, \theta_E^{(m)})$ are then obtained as

$$\begin{aligned} \frac{\partial Q(\theta_E, \theta_E^{(m)})}{\partial e_j} &= \sum_{i=1}^I K_i x_{ij} \\ &- \sum_{i=1}^I \left(K_i x_{ij} t_i^* \exp \left(\sum_{j=0}^J e_j x_{ij} \right) \right), \quad j = 0, 1, \dots, J, \end{aligned}$$

and

$$\begin{aligned} \frac{\partial^2 Q(\theta_E, \theta_E^{(m)})}{\partial e_p \partial e_q} &= - \sum_{i=1}^I \left(K_i x_{ip} x_{iq} t_i^* \exp \left(\sum_{j=0}^J e_j x_{ij} \right) \right), \\ p, q &= 0, 1, \dots, J. \end{aligned}$$

Furthermore, the required conditional expectation in the E-step is straightforward:

$$t_i^* = E(t_{ik} | \mathbf{z}, \theta_E^{(m)}) = \frac{1}{\lambda_i^{(m)}} + \tau_i \left(1 - \frac{n_i}{K_i (1 - \exp(-\lambda_i^{(m)} \tau_i))} \right),$$

where $\lambda_i^{(m)} = \exp \left(\sum_{j=0}^J e_j^{(m)} x_{ij} \right)$.

Furthermore, when the MLEs of the model parameters $\hat{\theta}_E = (\hat{e}_0, \hat{e}_1, \dots, \hat{e}_J)$ are determined, we immediately have the MLEs of the mean lifetime and the reliability at a mission time t under normal operating conditions $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0J})$, as

$$\hat{\mu}_E = \hat{\lambda}_0^{-1} = \exp \left(- \sum_{j=0}^J \hat{e}_j x_{0j} \right).$$

and

$$\widehat{R_E}(t) = \exp(-\hat{\lambda}_0 t) = \exp\left(-\exp\left(\sum_{j=0}^J \hat{e}_j x_{0j}\right) t\right),$$

respectively.

52.4.2 Confidence Intervals

Asymptotic Confidence Intervals

To construct confidence intervals for parameters of interest, asymptotic confidence intervals are commonly used when sample sizes are sufficiently large. This requires the asymptotic variance-covariance matrix of the MLEs of the model parameters. Under the EM framework, the missing information principle, discussed by Louis [28], is often employed for obtaining the observed information matrix of the MLEs of the model parameters. Balakrishnan and Ling [8] observed that, in case of the lifetimes being all censored, the observed information matrix obtained by the use of the missing information principle is equivalent to that obtained from the observed likelihood function. In one-shot device testing data, the observed information matrix of the MLEs is the negative of the second-order derivatives of the observed log-likelihood function with respect to the model parameters, i.e.,

$$\mathbf{I}_{obs} = -\left[\frac{\partial^2 \ln(L(\boldsymbol{\theta}; \mathbf{z}))}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}'}\right]_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}.$$

The asymptotic variance-covariance matrix of the MLEs of the model parameters is then the inverse of the observed information matrix, i.e.,

$$\hat{\mathbf{V}}_{\boldsymbol{\theta}} = \mathbf{I}_{obs}^{-1}.$$

In addition, the asymptotic variance of the MLE of a function of the model parameters, namely, the mean lifetime and the reliability, can be obtained by the delta method that requires the first-order derivatives of the corresponding function with respect to the model parameters. For example, the asymptotic variance of the MLE of a parameter of interest, ϕ , is

$$\hat{\mathbf{V}}_{\phi} = P' \hat{\mathbf{V}}_{\boldsymbol{\theta}} P,$$

where $P = \partial \phi / \partial \boldsymbol{\theta}$ is a vector of the first-order derivatives of ϕ with respect to $\boldsymbol{\theta}$.

Subsequently, the estimated standard error $\widehat{se}(\hat{\phi}) = \sqrt{\hat{\mathbf{V}}_{\phi}}$ is readily obtained. Consequently, the $100(1 - \delta)\%$ asymptotic confidence interval for ϕ is given by

$$\left(\hat{\phi} - z_{\frac{\delta}{2}} \widehat{se}(\hat{\phi}), \hat{\phi} + z_{\frac{\delta}{2}} \widehat{se}(\hat{\phi})\right),$$

where z_{δ} is the $100(1 - \delta)$ -th percentile of the standard normal distribution. Truncation on the bounds of the confidence intervals for the reliability and the mean lifetime of the devices may be needed, since the reliability has to lie between 0 and 1 and the mean lifetime has to be positive.

Jackknifed Confidence Intervals

The Jackknife method is a systematic re-sampling technique, omitting one observation at a time from the original sample, and can also be employed for constructing confidence intervals. In the case of one-shot device testing data, it is efficient to implement the Jackknife method because the calculation would not take too much time through the Jackknife technique.

Following the Jackknife method presented in [18], suppose the MLE $\hat{\phi}$ is obtained from the original data. For $i = 1, 2, \dots, I$, a device failed at τ_i is deleted from the data, and so both n_i and K_i are reduced by 1. Then, the MLE of ϕ based on the reduced data is obtained, and denoted by $\hat{\phi}_F^{(-i)}$. Similarly, for the cases when a device tested successfully at τ_i is deleted from the original data, the MLE of ϕ based on the reduced data (by reducing only K_i by 1) is also obtained and denoted by $\hat{\phi}_S^{(-i)}$. From this set of estimates, the bias of the MLE is given by

$$bias(\hat{\phi}) = -\frac{n-1}{n} \sum_{i=1}^I \left(n_i (\hat{\phi} - \hat{\phi}_F^{(-i)}) + (K_i - n_i) (\hat{\phi} - \hat{\phi}_S^{(-i)}) \right),$$

where $n = \sum_{i=1}^I K_i$. Hence, the Jackknife estimator of ϕ is simply

$$\hat{\phi}_{JK} = \hat{\phi} - bias(\hat{\phi}) = n\hat{\phi} - (n-1)\bar{\hat{\phi}},$$

where $\bar{\hat{\phi}} = \sum_{i=1}^I \left(n_i \hat{\phi}_F^{(-i)} + (K_i - n_i) \hat{\phi}_S^{(-i)} \right) / n$. In addition, its standard deviation can be calculated as

$$\widehat{se}_{JK}(\hat{\phi}) = \sqrt{\frac{n-1}{n} \sum_{i=1}^I \left\{ n_i (\hat{\phi}_F^{(-i)} - \bar{\hat{\phi}})^2 + (K_i - n_i) (\hat{\phi}_S^{(-i)} - \bar{\hat{\phi}})^2 \right\}}.$$

Hence, given the bias-corrected estimate $\hat{\phi}_{JK}$, the $100(1 - \delta)\%$ Jackknifed confidence interval for ϕ is then given by

$$\left(\hat{\phi}_{JK} - z_{\frac{\delta}{2}} \widehat{se}_{JK}(\hat{\phi}), \hat{\phi}_{JK} + z_{\frac{\delta}{2}} \widehat{se}_{JK}(\hat{\phi})\right).$$

As before, the Jackknifed confidence intervals for the mean lifetime and the reliability under normal operating conditions could be outside the range of their supports when the sample size is small. Therefore, we may have to do corrections on the confidence intervals, if necessary.

Parametric Bootstrap Confidence Intervals

Parametric bootstrap technique is an alternative to generate a set of estimates to construct confidence intervals for parameters of interest. By repeatedly performing bootstrap simulations, the corresponding distribution of the estimates can be approximated. The confidence intervals can then be constructed based on the empirical distribution of the estimates. The confidence intervals for a parameter of interest, ϕ , can be constructed by the parametric bootstrap method as follows:

1. Determine the MLE, $\hat{\theta}$, from the original data.
2. Obtain bootstrap samples $\{n_i^*, i = 1, 2, \dots, I\}$ based on $\hat{\theta}$ and $\{\tau_i, K_i, \mathbf{x}_i, i = 1, 2, \dots, I\}$
3. Determine the MLE of ϕ based on $\{n_i^*, i = 1, 2, \dots, I\}$, denoted by $\hat{\phi}_{BT}^b$
4. Repeat steps 2 and 3 B times to obtain the MLEs $\hat{\phi}_{BT}^b, b = 1, 2, \dots, B$
5. Sort the MLEs $\hat{\phi}_{BT}^b$ in an ascending order, denoted by $\hat{\phi}_{BT}^{[b]}, b = 1, 2, \dots, B$

Then, the $100(1 - \delta)\%$ parametric bootstrap confidence interval for ϕ is given by

$$\left(\hat{\phi}_{\lfloor \frac{\delta}{2}(B+1) \rfloor}^b, \hat{\phi}_{\lceil (1-\frac{\delta}{2})(B+1) \rceil}^b \right).$$

Here, the bootstrap confidence intervals do not require any correction.

Approximate Confidence Intervals

Hoyle [21] discussed various transformations for developing suitable confidence intervals in case of estimates with skewed distributions. When the sample size is small, the MLEs of the mean lifetime and the reliability may not possess normal distributions, and consequently the confidence intervals constructed directly by the asymptotic method may not maintain the nominal level of confidence. Here, we adopt the logit transformation [43] and have $\hat{f} = \ln(\hat{R}(t)/(1 - \hat{R}(t)))$ to be asymptotically normally distributed with the corresponding standard deviation, determined by the delta method, as

$$\widehat{se}(\hat{f}) = \frac{\widehat{se}(\hat{R}(t))}{\hat{R}(t)(1 - \hat{R}(t))},$$

where $\widehat{se}(\hat{R}(t))$ is the estimated standard error of $\hat{R}(t)$. From the $100(1 - \delta)\%$ confidence interval for f given by

$$\left(\hat{f} - z_{\frac{\delta}{2}} \widehat{se}(\hat{f}), \hat{f} + z_{\frac{\delta}{2}} \widehat{se}(\hat{f}) \right),$$

we can immediately obtain an approximate $100(1 - \delta)\%$ confidence interval for the reliability $R(t)$ to be

$$\left(\frac{\hat{R}(t)}{\hat{R}(t) + (1 - \hat{R}(t))S}, \frac{\hat{R}(t)}{\hat{R}(t) + (1 - \hat{R}(t))S^{-1}} \right),$$

where $S = \exp\left(z_{\frac{\delta}{2}} \widehat{se}(\hat{f})\right)$.

In a similar manner, the transformation approach can also be used to construct confidence intervals for the mean lifetime, which avoids having a negative lower bound for the mean lifetime. In this approach, we assume that $\ln(\hat{\mu})$ is asymptotically normally distributed with the corresponding standard deviation as $\widehat{se}(\ln(\hat{\mu})) = \widehat{se}(\hat{\mu})/\hat{\mu}$, where $\widehat{se}(\hat{\mu})$ is the estimated standard error of $\hat{\mu}$. This approach results in an approximate $100(1 - \delta)\%$ confidence interval for the mean lifetime μ as

$$\left(\hat{\mu} \exp\left(-\frac{z_{\frac{\delta}{2}} \widehat{se}(\hat{\mu})}{\hat{\mu}}\right), \hat{\mu} \exp\left(\frac{z_{\frac{\delta}{2}} \widehat{se}(\hat{\mu})}{\hat{\mu}}\right) \right).$$

It should be noted that the computational work for the estimated standard deviations of the reliability and the mean lifetime have been discussed earlier in Sect. 52.4.2.

52.4.3 Model Mis-Specification Analysis

Model-based methods have become a key part of reliability analysis. The gamma and Weibull distributions are quite similar as they can effectively model lifetimes with increasing, decreasing, and constant hazard rates. Besides, both these models include the exponential distributions as a special case. For these reasons, both these distributions are commonly used in reliability literature to model lifetime data. When the distribution is correctly chosen to fit one-shot device data, the MLEs possess less bias and smaller root mean square errors. In addition, the confidence intervals for parameters of interest are satisfactory in most cases. However, one has to be cautious about the improper fitting of distributions to data. In this regard, much attention has also been paid to model mis-specification analysis. This section examines the effects of model mis-specification on point estimation and likelihood inference.

Now, let us suppose the lifetimes of one-shot devices follow the gamma distributions but are wrongly assumed to have come from the Weibull distributions. Let θ_w^* be the quasi-MLE (QMLE) of the model parameters under the gamma distributions, given by (see [46])

$$\theta_w^* = \underset{\theta_w}{\operatorname{argmin}}(-\ln(L(\theta_w))).$$

To evaluate the effect of model mis-specification, let us denote the bias of the estimator of the parameter of interest ϕ_w under normal operating conditions \mathbf{x}_0 under the

Weibull distributions by $B_G(\hat{\phi}_W)$, which is equivalent to the parameter, ϕ_G , under the true gamma distributions. It can be expressed as

$$B_G(\hat{\phi}_W) = E[\hat{\phi}_W(\theta_W^*)] - \phi_G(\theta_G).$$

Then, the relative bias is given by

$$\kappa_G = \frac{B_G(\hat{\phi}_W)}{\phi_G(\theta_G)}.$$

Similarly, suppose the lifetimes of one-shot devices follow the Weibull distributions but are wrongly assumed to have come from the gamma distributions. Then, let θ_G^* be the QMLE of the model parameter under the Weibull distributions given by

$$\theta_G^* = \underset{\theta_G}{\operatorname{argmin}}(-\ln(L(\theta_G))).$$

Let $B_W(\hat{\phi}_G)$ denote the bias of the estimator of the parameter of interest ϕ_G under normal operating conditions \mathbf{x}_0 under the gamma distributions, which is equivalent to the parameter ϕ_W under the true Weibull distributions. It can be expressed as

$$B_W(\hat{\phi}_G) = E[\hat{\phi}_G(\theta_G^*)] - \phi_W(\theta_W).$$

Then, the relative bias is given by

$$\kappa_W = \frac{B_W(\hat{\phi}_G)}{\phi_W(\theta_W)}.$$

The model mis-specification issue on the likelihood inference was also investigated, when the distribution is wrongly specified. To measure the effect of model mis-specification, let $CP_G(\hat{\phi}_W)$ denote the coverage probability of the approximate confidence interval for a particular parameter of interest ϕ_W under the Weibull distributions, which is equivalent to ϕ_G under the true gamma distributions. It can be expressed as

$$CP_G(\hat{\phi}_W) = Pr(\phi_G \in (L_W, U_W)),$$

where L_W and U_W are the lower and upper bounds of the approximate confidence interval for ϕ_W , respectively. Similarly, let $CP_W(\hat{\phi}_G)$ denote the coverage probability of the approximate confidence interval for a particular parameter of interest ϕ_G under the gamma distributions, which is equivalent to ϕ_W under the true Weibull distributions. It can be expressed as

$$CP_W(\hat{\phi}_G) = Pr(\phi_W \in (L_G, U_G)),$$

where L_G and U_G are the lower and upper bounds of the approximate confidence interval for ϕ_G , respectively.

The numerical computations in [24] presented the relative bias of the estimators of the mean lifetime and the reliability for various settings and revealed that the estimation of the cdf for each test group is satisfactory even when the underlying distribution is mis-specified. The consequence of model mis-specification on point estimation is not serious when the normal operating conditions \mathbf{x}_0 are close to the elevated stress levels $\{\mathbf{x}_i, i = 1, 2, \dots, I\}$. However, due to the linkage between stress levels and lifetimes, the estimation of the mean lifetime and the reliability under \mathbf{x}_0 becomes imprecise, especially when the elevated stress levels are away from normal operating conditions. This observation suggests that having elevated stress levels near the normal operating conditions would minimize the effect of model mis-specification.

Furthermore, the performance of the approximate 95% confidence intervals for the mean lifetime and the reliability was examined. It was observed that the coverage probabilities under the true distributions maintain the nominal level. Also, these results showed that the model mis-specification is not serious only when the gamma distributions are mis-specified as the Weibull distributions in the case of small sample sizes. In general, the confidence intervals produce deflated coverage probabilities. Again, the consequence of model mis-specification on the likelihood inference is seen to be serious when the elevated stress levels are away from normal operating conditions, especially in the case of large sample sizes.

52.4.4 Model Selection

Due to the importance of system safety, it is of great interest in detecting if there is model mis-specification in the fitted model. Balakrishnan and Chimitova [4] compared several test statistics for testing the goodness-of-fit of assumed models: (1) a chi-square type statistic based on the difference between the empirical and expected numbers of failures at each inspection time, (2) Kolmogorov and (3) Cramer-von Mises-Smirnov type statistics based on the difference between the nonparametric MLE (NPMLE) of the lifetime distribution obtained from one-shot device testing data and the distribution specified under the null hypothesis, and (4) White type statistic comparing two estimators of the Fisher information.

Let R_i be the i -th residual for $i = 1, 2, \dots, I$. We define $n = \sum_{i=1}^I K_i$, $\hat{S}_0(t; \theta)$ as the baseline reliability function, $\hat{S}_n(t)$ as the NPMLE of the baseline reliability function, $n_i^* = K_i(1 - \hat{S}_n(R_i))$ as the empirical number of failures, and $\hat{n}_i = K_i(1 - \hat{S}_0(R_i; \hat{\theta}))$ as the expected number of failures.

The chi-square type statistic can be written as follows:

$$\chi_n^2 = \sum_{i=1}^I \frac{(n_i^* - \hat{n}_i)^2}{\hat{n}_i}.$$

It is clear that the values of the statistic strongly depend on the location of inspection times relative to the corresponding conditional distribution under test. It may happen that the value of the statistic will be small for some inappropriate distribution only because inspection times are located mainly in the right tail of the distribution. To avoid this problem, the weighted chi-square type statistic was introduced in the following form:

$$v\chi_n^2 = \sum_{i=1}^I \frac{v_{ij}(n_i^* - \hat{n}_i)^2}{\hat{n}_i},$$

where

$$v_{ij} = \frac{S_0(R_i; \hat{\theta})(1 - S_0(R_i; \hat{\theta}))}{\sum_{i=1}^I S_0(R_i; \hat{\theta})}.$$

The Kolmogorov type statistic is defined as

$$D_n = \sup_{0 < t < \infty} \left| \hat{S}_n(t) - S_0(t; \hat{\theta}_n) \right|.$$

The Cramer-von Mises-Smirnov type statistic is of the following form:

$$W_n^2 = \int_0^\infty \left(\hat{S}_n(t) - S_0(t; \hat{\theta}_n) \right)^2 dS_0(t; \hat{\theta}_n).$$

The White type statistic is based on the comparison of two estimators of the Fisher information matrix:

$$A_n(\hat{\theta}_n) = -\frac{1}{n} \sum_{i=1}^I \left[n_i \frac{\partial^2 \ln(1 - S_0(R_i; \hat{\theta}_n))}{\partial \lambda^2} + (K_i - n_i) \frac{\partial^2 \ln(S_0(R_i; \hat{\theta}_n))}{\partial \lambda^2} \right],$$

and

$$B_n(\hat{\theta}_n) = \frac{1}{n} \sum_{i=1}^I n_i \left(\frac{\partial \ln(1 - S_0(R_i; \hat{\theta}_n))}{\partial \lambda} \right) \left(\frac{\partial \ln(1 - S_0(R_i; \hat{\theta}_n))}{\partial \lambda} \right)' + \frac{1}{n} \sum_{i=1}^I (K_i - n_i) \left(\frac{\partial \ln(S_0(R_i; \hat{\theta}_n))}{\partial \lambda} \right) \left(\frac{\partial \ln(S_0(R_i; \hat{\theta}_n))}{\partial \lambda} \right)',$$

where λ is the vector of parameters of the baseline reliability function S_0 . The statistic is then defined as

$$V_n = \frac{|\det A_n(\hat{\theta}_n) - \det B_n(\hat{\theta}_n)|}{\det B_n(\hat{\theta}_n)}.$$

In addition, Ling and Balakrishnan [24] proposed two measures as specification tests for this purpose and then evaluated their performance. Akaike information criterion (AIC) (see [1, 2]) is commonly used to measure the relative quality of fits achieved by probability models for a given set of data. Let AIC_G and AIC_W denote the AIC values under the gamma and the Weibull distributions, respectively. For each probability distribution, there are $J + 1$ model parameters to be estimated for each of scale and shape parameters, and so AIC_G and AIC_W statistics are given by

$$AIC_G = 2(2J + 2) - 2 \ln(L(\hat{\theta}_G)) \quad \text{and}$$

$$AIC_W = 2(2J + 2) - 2 \ln(L(\hat{\theta}_W)).$$

When the assumed distribution does not fit the data, the logarithm of the likelihood value becomes small, and consequently the preferred model is the one with the smallest AIC value, and this is used for the purpose of model selection. Since the same number of parameters are in both gamma and Weibull distributions, the comparison of AIC values is exactly the same as the comparison of the log-likelihood values and also exactly the same as by the Bayesian information criterion (BIC). So, we can consider a specification test statistic as

$$D_{AIC} = AIC_G - AIC_W = 2(\ln(L(\hat{\theta}_W)) - \ln(L(\hat{\theta}_G))).$$

The gamma distributions are preferred when $D_{AIC} < 0$, while the Weibull distributions are preferred when $D_{AIC} > 0$.

Balakrishnan and Ling [8, 9] proposed a distance-based test statistic, which simply quantifies the distance between the observed and expected numbers of failures at each test group. Let M_G and M_W denote the values of the test statistic under the gamma and the Weibull distributions, respectively, which are given by

$$M_G = \max_i |n_i - K_i \hat{F}_G(\tau_i)| \quad \text{and} \quad M_W = \max_i |n_i - K_i \hat{F}_W(\tau_i)|.$$

So, when the assumed distribution does not fit the data well, we expect to observe a large value of the test statistic. We can then consider a specification test statistic as

$$D_M = M_G - M_W.$$

We can conclude that the gamma distributions are the preferred one when $D_M < 0$, while the Weibull distributions become the preferred one if $D_M > 0$.

We further define $(L, U) = (L_G, U_G)$ when $D_M < 0$ and $(L, U) = (L_W, U_W)$ when $D_M > 0$. When the specification test statistic D_M is implemented, the coverage probabilities of

the approximate confidence intervals for ϕ_G and ϕ_W can be expressed as

$$\begin{aligned} CP_G(\phi_G) &= Pr(\phi_G \in (L, U), D_M < 0) + Pr(\phi_G \in (L, U), D_M > 0) \\ &= Pr(\phi_G \in (L, U)|D_M < 0)P_G(D_M < 0) + Pr(\phi_G \in (L, U)|D_M > 0)P_G(D_M > 0), \end{aligned}$$

and

$$\begin{aligned} CP_W(\phi_W) &= Pr(\phi_W \in (L, U), D_M < 0) + Pr(\phi_W \in (L, U), D_M > 0) \\ &= Pr(\phi_W \in (L, U)|D_M < 0)P_W(D_M < 0) + Pr(\phi_W \in (L, U)|D_M > 0)P_W(D_M > 0), \end{aligned}$$

where $P_G(D_M < 0)$ represents the probability that the gamma distributions correctly selected and $P_W(D_M > 0)$ represents the probability that the Weibull distributions are correctly selected by the use of a specification test. Both $P_G(D_M < 0)$ and $P_W(D_M > 0)$ can be considered as the power of detection of mis-specified distributions by the specification test statistic. It is worth mentioning that the statistic D_M may be replaced by another specification statistic D_{AIC} .

A simulation study was carried out for evaluating the power of detection of model mis-specification for the AIC and the distance-based test statistic M for different settings. The obtained results revealed that the distance-based test statistic is slightly better than the AIC for detecting mis-specified distributions when the gamma distributions are the true one, while the AIC performs much better when the Weibull distributions are the true one. In practice, the true probability distributions are unknown; for this reason, the AIC is recommended as a specification test. Furthermore, the performance of the approximate 95% approximate confidence intervals for the mean lifetime and the reliability, with and without the use of a specification test, was evaluated. It was observed that, even when the coverage probabilities for mis-specified distributions are low in the case of large sample sizes, with the use of a specification test, the mis-specified distributions are unlikely to be selected as seen from the corresponding power values. Consequently, the coverage probabilities can be improved considerably and brought close to the nominal level, with the use of a model specification test.

On the other hand, likelihood ratio tests under generalized gamma distributions are useful for discriminating among those models. Firstly, we consider the likelihood ratio test for testing the null hypothesis $H_0 : v_j = 0, j = 0, 1, 2, \dots, J$ (gamma distributions) against the alternative hypothesis $H_1 : v$'s are not all equal to 0 (generalized gamma distributions). Under H_0 , the log-likelihood ratio test statistic is

$$LRT_G = -2(\ln(L(\hat{\theta}_G; \mathbf{z})) - \ln(L(\hat{\theta}_{GG}; \mathbf{z}))).$$

Similarly, we consider the likelihood ratio test for testing the null hypothesis $H_0 : u_j = 0, j = 0, 1, 2, \dots, J$ (Weibull distributions) against the alternative hypothesis $H_1 : u$'s are not all equal to 0 (generalized gamma distributions). Under H_0 , the log-likelihood ratio test statistic is

$$LRT_W = -2(\ln(L(\hat{\theta}_W; \mathbf{z})) - \ln(L(\hat{\theta}_{GG}; \mathbf{z}))).$$

The above null hypotheses are less likely to be true as the corresponding test statistic is large. Moreover, these test statistics are always positive and have an approximate chi-square distribution with degrees of freedom equal to $J + 1$ when the sample size is sufficiently large [47]. Therefore, we can compute the p-value to determine whether the gamma/Weibull distributions are appropriate to fit the observed data. For instance, if a p-value is less than 0.05, the corresponding distributions are not suggested to fit the data.

The performance of the likelihood ratio tests for choosing an appropriate model between the gamma and Weibull distributions to fit the observed data was determined in terms of percentages of rejection under the null hypothesis. The results showed that the likelihood ratio tests can successfully determine that Weibull distributions are inappropriate for fitting the data for all the considered sample sizes when the gamma distributions are the true one. Conversely, when the Weibull distributions are the correct one, the likelihood ratio tests performed satisfactory only when the sample size is sufficiently large. In conclusion, the Akaike information criterion and the distance-based test statistic and the likelihood ratio tests encountered the same challenge in one-shot device testing data.

52.4.5 Test of Proportional Hazard Rates

In the preceding section, the proportional hazards models have only the proportional hazards assumption and allow hazard rate to change in a nonparametric way. Here, a test for proportional hazards assumption based on one-shot device testing data is presented. Let us consider the distance-based test statistic [8, 9, 25]:

$$M = \max_i |n_i - \hat{n}_i|,$$

where $\hat{n}_i = K_i \hat{F}_{PH}(\tau_i)$ is the expected number of failures in the i -th test group under the proportional hazards models. The test statistic simply assesses the fit of the assumed models to the observed data, and so we would observe a large value of M when the assumed models are not good fit to the data.

Let $L_i = \lceil \hat{n}_i - M \rceil$ and $U_i = \lfloor \hat{n}_i + M \rfloor$. Under the proportional hazards models, we have $n_i \sim \text{Binomial}(K_i, \hat{F}_{PH}(\tau_i))$, and so, the corresponding exact p-value is given by

$$\text{p-value} = 1 - \prod_{i=1}^I \text{Pr}(L_i \leq n_i \leq U_i).$$

From the above expression, we can readily validate the proportional hazards assumption when the p-value is sufficiently large, i.e., p-value > 0.05 or 0.1, say.

52.4.6 Designs of CSALTs

Suppose that, in the i -th test group, K_i items are subject to J types of stress factors with higher-than-usual stress levels and inspected at v_i equally-spaced time points. Specifically, $K_{i,v}$ items are drawn and inspected at a specific time $\tau_{i,v}$, with $\sum_{v=1}^{v_i} K_{i,v} = K_i$. Then, $n_{i,v}$ failure items are collected from the inspection. Let us now assume that the lifetimes of devices in the i -th test group follow the Weibull distributions with scale parameter β_i and shape parameter η_i , wherein the scale and shape parameters are both related to the stress levels $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{ij})$ through the log-linear link functions.

To design CSALTs for one-shot devices, Balakrishnan and Ling [10] considered the optimization problem of determining the inspection frequency, the number of inspections at each condition, and the allocation of the devices by minimizing the asymptotic variance of the MLE of the reliability at a

mission time under normal operating conditions, subject to a specified budget and a termination time.

Suppose the budget for conducting CSALTs for one-shot devices, the operation cost in the i -th test group, the cost of devices (including the purchase of and testing cost), and the termination time are C_{budget} , $C_{oper,i}$, C_{item} , and τ_{ter} , respectively. Given a test plan, ζ , that involves the inspection frequency, f , the number of inspections in the i -th test group, $v_i \geq 2$, and the allocation of devices, $K_{i,v}$, for $i = 1, 2, \dots, I, v = 1, 2, \dots, v_i$, it is assumed that the total cost of implementing the test plan is

$$TC(\zeta) = C_{item} \sum_{i=1}^I \sum_{v=1}^{v_i} K_{i,v} + \sum_{i=1}^I C_{oper,i} v_i f,$$

and

$$\max_i v_i f \leq \tau_{ter}.$$

Balakrishnan and Ling [10] presented a procedure of determining CSALT plans for one-shot devices.

52.5 Minimum Density Power Divergence Estimation

Balakrishnan et al. [14, 15] developed a new family of estimators, the minimum density power divergence estimators (MDPDEs), for one-shot device testing data under the exponential and gamma distributions. They revealed that some MDPDEs have a better behavior than the MLE in relation to robustness.

52.5.1 Minimum Density Power Divergence Estimators

Consider the theoretical probability vector

$$p(\theta) = [F(\tau_1; \theta), 1 - F(\tau_1; \theta), \dots, F(\tau_I; \theta), 1 - F(\tau_I; \theta)]',$$

and the observed probability vector

$$\hat{p} = \left[\frac{n_1}{K_1}, \frac{K_1 - n_1}{K_1}, \dots, \frac{n_I}{K_I}, \frac{K_I - n_I}{K_I} \right]'$$

The Kullback-Leibler divergence between the probability vectors is given by

$$\begin{aligned} d_{KL}(p(\theta), \hat{p}) &= \frac{1}{I} \sum_{i=1}^I \left(\frac{n_i}{K_i} \ln \left(\frac{n_i}{K_i F(\tau_i; \theta)} \right) + \frac{K_i - n_i}{K_i} \ln \left(\frac{K_i - n_i}{K_i (1 - F(\tau_i; \theta))} \right) \right) \\ &= \frac{1}{I} (s - \ln(L(\theta; \mathbf{z}))) \end{aligned}$$

with s being a constant not dependent on θ . Hence, the MLE of θ can also be defined as

$$\hat{\theta} = \arg \min_{\theta} d_{KL}(p(\theta), \hat{p}).$$

Suppose $y_{ik} \sim \text{Bernoulli}(\pi_i(\theta))$, where $\pi_i(\theta) = F(\tau_i; \theta)$. The MDPDEs with turning parameter $\omega \geq 0$ is given by

$$\hat{\theta}_{\omega} = \arg \min_{\theta} V(y_{ik}, \omega),$$

$$V(y_{ik}, \omega) = (\pi_i(\theta))^{\omega+1} + (1 - \pi_i(\theta))^{\omega+1} - \left(1 + \frac{1}{\omega}\right) ((\pi_i(\theta))^{y_{ik}} (1 - \pi_i(\theta))^{1-y_{ik}})^{\omega}.$$

Here, the term $\pi_i^{y_{ik}}(\theta)(1 - \pi_i(\theta))^{1-y_{ik}}$ plays the role of the density in our context. The divergence measure to be minimized can be simplified as

$$\begin{aligned} \frac{1}{I} \sum_{i=1}^I \sum_{k=1}^{K_i} V(y_{ik}, \omega) &= \sum_{i=1}^I (\pi_i(\theta))^{\omega+1} + (1 - \pi_i(\theta))^{\omega+1} \\ &\quad - \sum_{i=1}^I \left(1 + \frac{1}{\omega}\right) \left(\binom{n_i}{K_i} (\pi_i(\theta))^{\omega} + \binom{K_i - n_i}{K_i} (1 - \pi_i(\theta))^{\omega} \right). \end{aligned}$$

From the above expression, there is an alternative for $\hat{\theta}_{\omega}$, in which only a divergence measure between two probabilities is involved. Given two probability vectors $p(\theta)$ and \hat{p} , the

density power divergence (DPD) measure between $p(\theta)$ and \hat{p} , with tuning parameter ω , is given by

$$\begin{aligned} d_{\omega}(p(\theta), \hat{p}) &= \sum_{i=1}^I (\pi_i(\theta))^{\omega+1} + (1 - \pi_i(\theta))^{\omega+1} \\ &\quad - \sum_{i=1}^I \left(1 + \frac{1}{\omega}\right) \left(\binom{n_i}{K_i} (\pi_i(\theta))^{\omega} + \binom{K_i - n_i}{K_i} (1 - \pi_i(\theta))^{\omega} \right) \\ &\quad + \sum_{i=1}^I \left(\frac{1}{\omega}\right) \left(\binom{n_i}{K_i}^{\omega+1} + \binom{K_i - n_i}{K_i}^{\omega+1} \right). \end{aligned}$$

We observe that the last term $(n_i/K_i)^{\omega+1} + ((K_i - n_i)/K_i)^{\omega+1}$ does not have any role in the minimization of $d_{\omega}(\hat{p}, p(\theta))$, and the MDPDEs can alternatively be defined as

$$\hat{\theta}_{\omega} = \arg \min_{\theta} d_{\omega}(\hat{p}, p(\theta)).$$

For $\omega = 0$, we have

$$d_{\omega=0}(\hat{p}, p(\theta)) = \lim_{\omega \rightarrow 0^+} d_{\omega}(\hat{p}, p(\theta)) = d_{KL}(\hat{p}, p(\theta)).$$

This result implies that the MDPDEs contains the MLE as a particular case.

52.5.2 Wald-Type Tests

Balakrishnan et al. [14, 15] also presented Wald-type test statistics for testing different hypotheses concerning the parameter θ . Consider the composite null hypotheses of the type

$$m(\theta) = \mathbf{0}_r, \quad \text{and} \quad M(\theta) = \frac{\partial m(\theta)'}{\partial \theta}$$

with rank $M(\theta) = r$. The Wald-type test statistic is

$$W_n(\hat{\theta}_{\omega}) = nm(\hat{\theta}_{\omega})' \left(M(\hat{\theta}_{\omega})' \Sigma(\hat{\theta}_{\omega}) m(\hat{\theta}_{\omega}) \right)^{-1} m(\hat{\theta}_{\omega}),$$

One may refer to [14, 15] for the computation of $\Sigma(\hat{\theta}_{\omega})$ under the exponential and gamma distributions. Subsequently, at δ significance level, we decide to reject the null hypothesis when

$$W_n(\hat{\theta}_{\omega}) > \chi_{r, \delta}^2.$$

Besides, let

$$\ell_{\omega}(\theta_1, \theta_2) = m(\theta_1)' \left(M(\theta_2)' \Sigma_{\omega}(\theta_2) m(\theta_2) \right)^{-1} m(\theta_1),$$

and

$$\sigma_{W_n, \omega}(\theta^*) = \left. \frac{\partial \ell_\omega(\theta, \theta^*)}{\partial \theta'} \right|_{\theta=\theta^*} \Sigma_\omega(\theta^*) \left. \frac{\partial \ell_\omega(\theta, \theta^*)}{\partial \theta} \right|_{\theta=\theta^*}.$$

We can give an approximation of the power function of the Wald-type test statistic in θ^* to be

$$\begin{aligned} \pi_{W_n}(\theta^*) &= Pr\left(W_n(\hat{\theta}_\omega) > \chi_{r,\delta}^2\right) \\ &= 1 - \Phi_n\left(\frac{1}{\sigma_{W_n, \omega}(\theta^*)} \left(\frac{\chi_{r,\delta}^2}{\sqrt{n}} - \sqrt{n} \ell_\omega(\hat{\theta}_\omega, \theta^*)\right)\right), \end{aligned}$$

where $\Phi_n(x)$ tending uniformly to the standard normal distribution $\Phi(x)$.

They observed that the robustness of the MDPDE seems to increase with increasing ω ; their efficiency in case of pure data decreases slightly. Same happens with the Wald-type tests. A moderate value of ω is expected to provide the best trade-off for possibly contaminated data, but a data-driven choice of ω would be more convenient. A useful procedure for the data-based selection of ω was proposed by Warwick and Jones [45].

52.6 One-Shot Devices with Multiple Failure Modes

Modern devices are generally composed of multiple components, and it is common for devices to fail because of one of these components being failed. For instance, fire extinguishers contain liquid carbon dioxide, valves, safety fuses, and airbags contain crash sensors, inflators, and compressed gas. If the failed components can be found after tests, multivariate distributions can then be used to model the lifetime distribution of each type of components as well as the lifetime distribution of the devices.

52.6.1 Competing Risks Models for Independent Failure Models

Balakrishnan et al. [11–13] analyzed one-shot device testing data under competing risks models with the exponential and the Weibull distributions. CSALTs with I test groups for one-shot devices with M failure modes are set up as follows:

For $i = 1, 2, \dots, I$, in the i -th test group,

1. K_i devices are subject to elevated stress levels \mathbf{x}_i .
2. These devices are only inspected at time τ_i .
3. The number of devices with the m -th failure mode is denoted by n_{mi} , for $m = 1, 2, \dots, M$.

Let us now assume that there are only two failure modes and denote the random variable for the time to the m -th failure mode by T_{mik} , for $m = 1, 2, i = 1, 2, \dots, I$, and $k = 1, 2, \dots, K_i$, respectively. In this work, it is assumed that T_{mik} are independent of each other and that at most one failure mode can be observed in each device. Furthermore, Δ_{ik} is defined to be the indicator for the k -th device under in the i -th test group. If the device functions, we will set $\Delta_{ik} = 0$. However, if the device does not function, we will identify (by careful inspection) the specific failure mode. If the m -th failure mode is observed, we will denote this event by $\Delta_{ik} = m$, for $m = 1, 2$. Mathematically, the indicator Δ_{ik} is then defined as

$$\Delta_{ik} = \begin{cases} 0 & \text{if } \min(T_{1ik}, T_{2ik}) > \tau_i \\ 1 & \text{if } T_{1ik} < \min(T_{2ik}, \tau_i) \\ 2 & \text{if } T_{2ik} < \min(T_{1ik}, \tau_i) \end{cases}$$

and then δ_{ik} will be used for the realization of Δ_{ik} . Furthermore, we denote p_{0i} , p_{1i} , and p_{2i} for the reliability, the probability of observing the 1st failure mode, and the probability of observing the 2nd failure mode, respectively. Then, the likelihood function is given by

$$L(\theta; \mathbf{z}) = \prod_{i=1}^I \prod_{m=0}^2 (p_{mi}(\theta))^{n_{mi}},$$

where $n_{0i} = K_i - n_{1i} - n_{2i}$.

Balakrishnan et al. [11, 13] developed EM algorithms to find the MLEs of the model parameters, the mean lifetime, and the reliability under normal operating conditions. In addition, they measured the goodness-of-fit of the assumed model using a distance-based test statistic of the form

$$U = \max_i (|n_{0i} - \hat{n}_{0i}|, |n_{1i} - \hat{n}_{1i}|, |n_{2i} - \hat{n}_{2i}|),$$

where \hat{n}_{mi} is the expected numbers of n_{mi} based on the assumed model for $m = 0, 1, 2, i = 1, 2, \dots, I$. The statistic U quantifies the distance between the fitted model and the observed data. If the assumed model does not fit the data, the distance will be large. If the assumed model is true, $\{\hat{n}_{mi}, m = 0, 1, 2, i = 1, 2, \dots, I\}$, then follow the multinomial distribution with probabilities close to $\{p_{mi}, m = 0, 1, 2, i = 1, 2, \dots, I\}$. Consequently, the exact p -value of the test statistic U can be found as follows:

$$p\text{-value} = 1 - \prod_i \left(\sum_{n_{2i}=b_{2i}^u}^{b_{2i}^l} \sum_{n_{1i}=b_{1i}^u(n_{2i})}^{b_{1i}^l(n_{2i})} \frac{K_i!}{n_{0i}! n_{1i}! n_{2i}!} \hat{p}_{0i}^{n_{0i}} \hat{p}_{1i}^{n_{1i}} \hat{p}_{2i}^{n_{2i}} \right),$$

where $b_{1i}^l(n_{2i}) = \max(0, \lceil \hat{n}_{1i} - U \rceil, \lceil \hat{n}_{1i} + \hat{n}_{2i} - U \rceil - n_{2i})$, $b_{1i}^u(n_{2i}) = \min(K_i - n_{2i}, \lceil \hat{n}_{1i} + U \rceil, \lceil \hat{n}_{1i} + \hat{n}_{2i} + U \rceil - n_{2i})$, $b_{2i}^l = \max(0, \lceil \hat{n}_{2i} - U \rceil)$ and $b_{2i}^u = \min(K_i, \lceil \hat{n}_{2i} - U \rceil)$ for

$i = 1, 2, \dots, I$. If the exact p-value is smaller than the desired level, say, 0.05 or 0.1, then we may conclude that the assumed model does not fit the data well.

52.6.2 Copula Models for Correlated Failure Models

Subsequently, Ling et al. [26] considered two Archimedean copula models—the Gumbel-Hougaard copula and Frank copula—for analyzing one-shot device data with two correlated failure modes, to relax the independence assumption between two failure modes. The concept of copulas was first introduced by Sklar [40]. Copula models have been widely used to capture the dependence between random variables.

Consider one-shot devices with only two failure modes under CSALTs with I higher-than-normal stress levels, each of which is subject to an accelerating factor, and with ν inspection times. For $i = 1, 2, \dots, I, \nu = 1, 2, \dots, \nu, K_{i,\nu}$ devices are placed at stress level x_i and inspected at inspection time τ_ν . The numbers of devices without failures ($n_{i,\nu,0}$), with only the 1st failure mode ($n_{i,\nu,1}$), and with only the 2nd failure mode ($n_{i,\nu,2}$) are recorded. The one-shot device testing data collected at stress level x_i and at inspection time τ_ν can thus be summarized below in Table 52.2.

For notational convenience, $\mathbf{z} = \{x_i, \tau_j, K_{i,\nu}, n_{i,\nu,0}, n_{i,\nu,1}, n_{i,\nu,2}, i = 1, 2, \dots, I, \nu = 1, 2, \dots, \nu\}$ is denoted for the observed data. It is worth noting that both failure modes 1 and 2 may be observed at the inspection time and the number of devices with both failure modes is denoted as $n_{i,\nu,12}$, where $K_{i,\nu} = n_{i,\nu,0} + n_{i,\nu,1} + n_{i,\nu,2} + n_{i,\nu,12}$, for $i = 1, 2, \dots, I, \nu = 1, 2, \dots, \nu$. Moreover, the total number of devices with failure mode m is denoted as $N_{i,\nu,m} = n_{i,\nu,m} + n_{i,\nu,12}$, for $m = 1, 2$. In one-shot device testing data with two failure modes, let $T_{i,m}$ denote the time to the m -th failure mode at stress level x_i , where $i = 1, 2, \dots, I, m = 1, 2$. Suppose that $T_{i,m}$ has a distribution with the marginal probability distribution $F_{i,m}(t)$.

Gumbel-Hougaard copula that can characterize the positive dependence among random variables is one popular copula model with a range of applications. It is further assumed that the correlation between times to two failure modes changes over the stress levels, and under the Gumbel-

Table 52.2 Contingency table showing one-shot device testing data with two failure modes

Stress level x_i , Inspection time τ_ν			
Failure mode 2	Failure mode 1		Total
	Presence	Absence	
Presence	$n_{i,\nu,12}$	$n_{i,\nu,2}$	$N_{i,\nu,2}$
Absence	$n_{i,\nu,1}$	$n_{i,\nu,0}$	$K_{i,\nu} - N_{i,\nu,2}$
Total	$N_{i,\nu,1}$	$K_{i,\nu} - N_{i,\nu,1}$	$K_{i,\nu}$

Hougaard copula, $\ln(\alpha_i - 1)$ relates to stress level x_i in a linear form, namely,

$$y_i = \ln(\alpha_i - 1) = a_0 + a_1 x_i, \quad i = 1, 2, \dots, I.$$

Then, the joint cdf of $T_{i,1}$ and $T_{i,2}$ under the Gumbel-Hougaard copula can be written as

$$\begin{aligned} C_{\alpha_i}(t_1, t_2) &= P(T_{i,1} \leq t_1, T_{i,2} \leq t_2) \\ &= \exp\left(-\left[(-\log F_{i,1}(t_1))^{\alpha_i} + (-\log F_{i,2}(t_2))^{\alpha_i}\right]^{1/\alpha_i}\right). \end{aligned}$$

It is worth noting that Kendall’s tau, defined as the probability of concordance minus the probability of discordance of two random variables, is a measure of the degree of correspondence between two random variables. Kendall’s tau under the Gumbel-Hougaard copula can be expressed in terms of the dependence parameter as follows:

$$\Lambda_i = 1 - \frac{1}{\alpha_i}.$$

The Frank copula is another popular copula model because these copulas allow to model positive and negative dependence among random variables. Here, the dependence parameter is assumed to relate to stress level x_i in a linear form, namely,

$$y_i = \alpha_i = a_0 + a_1 x_i, \quad i = 1, 2, \dots, I.$$

Then, the joint cdf of $T_{i,1}$ and $T_{i,2}$ under the Frank copula is thus given by

$$\begin{aligned} C_{\alpha_i}(t_1, t_2) &= P(T_{i,1} \leq t_1, T_{i,2} \leq t_2) \\ &= -\frac{1}{\alpha_i} \ln\left(1 + \frac{(\exp(-\alpha_i F_{i,1}(t_1)) - 1)(\exp(-\alpha_i F_{i,2}(t_2)) - 1)}{(\exp(-\alpha_i) - 1)}\right). \end{aligned}$$

In addition, the Kendall’s tau can also be approximated by the dependence parameter as follows:

$$\Lambda_i \approx 1 - \frac{4}{\alpha_i} \left(1 - \frac{\pi^2}{6\alpha_i}\right).$$

The likelihood function of $\theta = (a_0, a_1)$ is then given by

$$L(\theta; \mathbf{z}) = \sum_{i=1}^I \sum_{v=1}^v n_{i,v,12} \ln(C_\theta(i, v)) + n_{i,v,1} \ln(\hat{F}_{i,1}(\tau_v) - C_\theta(i, v)) + n_{i,v,2} \ln(\hat{F}_{i,2}(\tau_v) - C_\theta(i, v)) + n_{i,v,0} \ln(1 - \hat{F}_{i,1}(\tau_v) - \hat{F}_{i,2}(\tau_v) + C_\theta(i, v)),$$

where $C_\theta(i, v) = C_{\alpha_i(\theta)}(\tau_v, \tau_v)$ and $\hat{F}_{i,m}(\tau_v) = N_{i,v,m}/K_{i,v}$.

It is observed that $\hat{F}_{i,s}(\tau_v) = C_\theta(i, v)$ when $\hat{F}_{i,m}(\tau_v) = 1$ for $s \neq m$, which leads to the term $\ln(\hat{F}_{i,s}(\tau_v) - C_\theta(i, v))$ in the log-likelihood function being undefined. To overcome this challenge, the replacement of 1 by $K_{i,v}(0.99^{(1/K_{i,v})})$ on $\hat{F}_{i,m}(\tau_v)$ is made as an adjustment. Similarly, for the Gumbel-Hougaard copula, $C_\theta(i, v)$ is undefined when $\hat{F}_{i,m}(\tau_v) = 0$, and thus the replacement of 0 by $K_{i,v}(1 - 0.99^{(1/K_{i,v})})$ on $\hat{F}_{i,m}(\tau_v)$ is also made as an adjustment. In short, an appropriate adjustment is required when either (i) $N_{i,v,m} = 0$ or (ii) $N_{i,v,m} = K_{i,v}$, for $m = 1, 2, i = 1, 2, \dots, I, v = 1, 2, \dots, v$.

Ling et al. [26] showed that one-shot devices with independent failure modes have shorter (longer) lifetimes than those with highly and positively (negatively) correlated failure modes. The Archimedean copula family contains a large variety of copulas that represent different dependence structures. Interested readers may refer to [34].

52.7 Step-Stress Accelerated Life Tests

Compared with CSALTs, in which each device is subject to only one pre-specific stress level, SSALTs wherein stress levels increase step-by-step over time [29] would require less samples and be more efficient and less costly to collect lifetime data. Thus, SSALTs have attracted great attention in literature. Ling [23] and Ling and Hu [27] studied simple SSALTs, which contain only two stress levels, under the exponential and the Weibull distributions.

Consider simple SSALTs wherein the stress levels changed only once from the test. Suppose that $0 < \tau_1 < \tau_2, 0 < K_1 < n$, and $x_1 < x_2$ and that all n devices are exposed to the same initial stress level x_1 . K_1 devices are selected to be tested at a pre-specified inspection time τ_1 , the number of failures n_1 are recorded. Then, the stress level increased to x_2 . All the remaining $K_2 = n - K_1$ devices are to be tested at another pre-specified inspection time τ_2 , and the number of failures n_2 are recorded. The one-shot device testing data thus observed can be summarized below in Table 52.3. Given one-shot device testing data, $\mathbf{z} = \{\tau_i, K_i, n_i, x_i, i = 1, 2\}$. In one-shot device testing data under simple SSALTs, the likelihood function is then given by

Table 52.3 One-shot device testing data under simple SSALTs

Stage	Inspection time	# of tested devices	# of failures	Stress level
1	τ_1	K_1	n_1	x_1
2	τ_2	K_2	n_2	x_2

$$L(\theta; \mathbf{z}) = \prod_{i=1}^2 (1 - R(\tau_i; \theta))^{n_i} (R(\tau_i; \theta))^{K_i - n_i}.$$

52.7.1 Exponential Models

Let T denote the lifetime of devices that follows the exponential distributions. Under the cumulative exposure model [35, 36], the cumulative hazard function, the reliability function, and the pdf are given by

$$H(t) = \begin{cases} \alpha_1 t, & 0 < t \leq \tau_1 \\ \alpha_1 \tau_1 + \alpha_2 (t - \tau_1), & t > \tau_1 \end{cases},$$

$$R(t) = \exp(-H(t)) = \begin{cases} \exp(-\alpha_1 t), & 0 < t \leq \tau_1 \\ \exp(-(\alpha_1 \tau_1 + \alpha_2 (t - \tau_1))), & t > \tau_1 \end{cases},$$

and

$$f(t) = -R'(t) = \begin{cases} \alpha_1 \exp(-\alpha_1 t), & 0 < t \leq \tau_1 \\ \alpha_2 \exp(-(\alpha_1 \tau_1 + \alpha_2 (t - \tau_1))), & t > \tau_1 \end{cases},$$

where $\alpha_1 > 0$ and $\alpha_2 > 0$ are the rate parameters at stages 1 and 2, respectively. We further assume that the rate parameters are related to the stress level in a log-linear form as

$$\alpha_i = \exp(a_0 + a_1 x_i).$$

Here, we denote $\theta_E = \{a_0, a_1\}$ as the model parameters to be estimated. Ling [23] developed an EM algorithm to find the MLEs of the model parameters. Furthermore, the mean lifetime under the normal operating condition x_0 is given by

$$\mu_E = \alpha_0^{-1} = \exp(-a_0 - a_1x_0).$$

In addition, let $\pi = K_1/n$, $A_i = R(\tau_i)^{-1} + (1 - R(\tau_i))^{-1}$ and $X_{ik} = \partial R(\tau_i)/\partial a_k$, $r_{kk} = A_2X_{2k}^2$, $r_{10} = A_2X_{21}X_{20}$, $s_{kk} = A_1X_{1k}^2 - A_2X_{2k}^2$, $s_{10} = A_1X_{11}X_{10} - A_2X_{21}X_{20}$. The estimated standard error of the mean lifetime under the normal operating condition is

$$\widehat{se}(\hat{\mu}_E) = \frac{(r_{11} - 2r_{10}x_0 + r_{00}x_0^2) + (s_{11} - 2s_{10}x_0 + s_{00}x_0^2)\pi}{n\alpha_0^2(r_{00}s_{11} + r_{11}s_{00} - 2r_{10}s_{10})\pi(1 - \pi)}.$$

It is noting that \hat{a}_0 and \hat{a}_1 are the MLEs of the model parameters of a_0 and a_1 , it follows that $\hat{\theta} \sim N_2(\theta, V_\theta)$, where

$$\theta = \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} \quad \text{and} \quad V_\theta = \begin{pmatrix} \sigma_0^2 & \sigma_0\sigma_1\rho \\ \sigma_0\sigma_1\rho & \sigma_1^2 \end{pmatrix}.$$

Then, it can be easily seen that the logarithm of the estimated mean lifetime under the normal operating condition, $\log(\hat{\mu}_E) = \hat{a}_0 + \hat{a}_1x_0$, is asymptotically normal distributed.

52.7.2 Weibull Models

Subsequently, Ling and Hu [27] extended this work to the Weibull distributions with constant shape parameter and varying scale parameter. Again, under the cumulative exposure model, the cumulative hazard function and the reliability function are derived as

$$H(t) = \begin{cases} \left(\frac{t}{\alpha_1}\right)^{a_2}, & 0 < t \leq \tau_1 \\ \left(\frac{\tau_1}{\alpha_1} + \frac{t-\tau_1}{\alpha_2}\right)^{a_2}, & t > \tau_1 \end{cases},$$

and

$$R(t) = \exp(-H(t)) = \begin{cases} \exp\left(-\left(\frac{t}{\alpha_1}\right)^{a_2}\right), & 0 < t \leq \tau_1 \\ \exp\left(-\left(\frac{\tau_1}{\alpha_1} + \frac{t-\tau_1}{\alpha_2}\right)^{a_2}\right), & t > \tau_1 \end{cases},$$

where $\alpha_1 > 0$ and $\alpha_2 > 0$ are the scale parameters at the first and second stages, respectively, and $a_0 > 0$ is the common shape parameter. It is further assumed that the scale parameters relate to stress levels in a log-linear link function, i.e.,

$$\alpha_i = \exp(a_0 + a_1x_i).$$

Then, the reliability at a mission time t under the normal operating condition x_0 can be expressed as

$$\begin{aligned} R_0(t) &= \exp\left(-\left(\frac{t}{\alpha_0}\right)^{a_2}\right) \\ &= \exp(-\exp(-a_2(a_0 + a_1x_0 - \log(t))))). \end{aligned}$$

For notational convenience, we define $\theta_W = \{a_0, a_1, a_2\}$ as the model parameters to be estimated.

However, in this setting, the determinant of the 3×3 expected Fisher information matrix equals zero, implying that the matrix is singular and the inverse of the matrix does not exist. As a result, the asymptotic variance-covariance matrix of the MLEs of the model parameters cannot be obtained. Here, the matrix is singular because the model parameters $\mathbf{a} = \{a_0, a_1, a_2\}$ are not identifiable under the data structure presented in Table 52.3. SSALTs with more than two stress levels under Weibull distributions will be investigated.

52.7.3 Designs of Simple SSALTs

Ling [23] and Ling and Hu [27] presented procedures to decide the sample size and the inspection times for simple SSALTs, given the model parameters, the standard error, the stress levels, and the normal operating condition.

Consider simple SSALTs for one-shot devices under the exponential distributions; Ling [23] considered the optimal designs that minimize the standard error of the mean lifetime. The optimal proportion of devices inspected at τ_1 is

$$\pi_E = \arg \min_{0 < \pi < 1} \widehat{se}(\hat{\mu}_E) = \left(1 + \sqrt{\frac{A_1(X_{11} - X_{10}x_0)^2}{A_2(X_{21} - X_{20}x_0)^2}}\right)^{-1}.$$

Subsequently, Ling and Hu [27] considered simple SSALTs for one-shot devices under the Weibull distributions with constant shape parameter. Since the asymptotic variance-covariance matrix of the MLEs of the model parameters cannot be obtained, two scenarios were discussed: (1) a known shape parameter a_2 and (2) a known parameter about stress level a_1 .

If a_2 is known, then the optimal proportion of devices inspected at τ_1 is

$$\pi_{W1} = \left(1 + \left(\frac{X_{11}}{X_{21}}\right) \sqrt{\frac{A_1}{A_2}}\right)^{-1}.$$

It is worth noting that the percentage involves X_{21} , which relies on (x_1, τ_1, τ_2) , but is independent of the mission time t .

On the other hand, when the mean lifetimes, $E[T_i] < E[T_j]$, at two standardized stress levels, $0 < x_j < x_i < 1$, can be suggested by experts or obtained from previous similar studies, the parameter can also be obtained. Let T_m be the lifetime having the Weibull distributions with shape parameter a_2 and scale parameter $\alpha_m = \exp(a_0 + a_1x_m)$, where $m = \{i, j\}$. The mean lifetime is therefore given by

$$E[T_m] = \alpha_m \Gamma \left(1 + \frac{1}{a_2} \right),$$

where $\Gamma(\cdot)$ is the gamma function. It follows that

$$\log \left(\frac{E[T_i]}{E[T_j]} \right) = \log \left(\frac{\alpha_i}{\alpha_j} \right) = a_1(x_i - x_j).$$

Hence, the parameter about stress level can be expressed as

$$a_1 = \frac{\log(E[T_i]) - \log(E[T_j])}{x_i - x_j}.$$

It is therefore reasonable to fix a_1 as a known parameter. Hence, the optimal proportion of devices inspected at τ_1 is

$$\pi_{w2} = \left(1 + \left(\frac{a_2 X_{12} + d_2 X_{10}}{a_2 X_{22} + d_2 X_{20}} \right) \sqrt{\frac{A_1}{A_2}} \right)^{-1},$$

where $d_2 = \log(t) - a_0$.

In this setup, the proportion is not independent of the mission time t . Also, it is revealed that the asymptotic standard deviation is minimized when either $\pi_{w2} = 1$ or $\pi_{w2} = 0$. It thus suggested increasing the stress level once and all the devices are tested at the termination time. It is worth noting that stress levels affect the inspection times but have no effect on sample size. Most importantly, in this setup where the parameter about stress level a_1 is known, the sample size is relatively small, and the simulated standard deviation is relatively robust, compared to the preceding scenario where the shape parameter is known. It suggested that fixing the parameter about stress level is more preferable than fixing the shape parameter.

Acknowledgments Our sincere thanks go to the editors of this volume for sending an invitation to us which provided an impetus for preparing this overview article. This work was supported by grants from the Education University of Hong Kong (Ref. RG22/2018-2019R, IDS-2 2019) and the Research Grants Council of the Hong Kong Special Administrative Region, China (Ref. 28300114).

References

1. Akaike, H.: On entropy maximization principle. In: Krishnaiah, P.R. (ed.) *Applications of Statistics*, pp. 27–41. North-Holland, Amsterdam (1977)
2. Akaike, H.: Likelihood of a model and information criteria. *J. Econ.* **16**(1), 3–14 (1981)
3. Bain, L.J., Engelhardt, M.: Reliability test plans for one-shot devices based on repeated samples. *J. Qual. Technol.* **23**(4), 304–311 (1991)
4. Balakrishnan, N., Chimitova, E.: Goodness-of-fit tests for one-shot device accelerated life testing data. *Commun. Stat. Simul. Comput.* **46**(5), 3723–3734 (2017)
5. Balakrishnan, N., Rao, C.R.: *Handbook of Statistics: Advances in Reliability*. Elsevier, New York (2001)
6. Balakrishnan, N., Ling, M.H.: EM algorithm for one-shot device testing under the exponential distribution. *Comput. Stat. Data Anal.* **56**(3), 502–509 (2012)
7. Balakrishnan, N., Ling, M.H.: Multiple-stress model for one-shot device testing data under exponential distribution. *IEEE Trans. Reliab.* **61**(3), 809–821 (2012)
8. Balakrishnan, N., Ling, M.H.: Expectation maximization algorithm for one shot device accelerated life testing with Weibull lifetimes, and variable parameters over stress. *IEEE Trans. Reliab.* **62**(2), 537–551 (2013)
9. Balakrishnan, N., Ling, M.H.: Gamma lifetimes and one-shot device testing analysis. *Reliab. Eng. Syst. Saf.* **126**, 54–64 (2014)
10. Balakrishnan, N., Ling, M.H.: Best constant-stress accelerated life-test plans with multiple stress factors for one-shot device testing under a Weibull distribution. *IEEE Trans. Reliab.* **63**(4), 944–952 (2014)
11. Balakrishnan, N., So, H.Y., Ling, M.H.: EM algorithm for one-shot device testing with competing risks under exponential distribution. *Reliab. Eng. Syst. Saf.* **137**, 129–140 (2015)
12. Balakrishnan, N., So, H.Y., Ling, M.H.: A Bayesian approach for one-shot device testing with exponential lifetimes under competing risks. *IEEE Trans. Reliab.* **65**(1), 469–485 (2016)
13. Balakrishnan, N., So, H.Y., Ling, M.H.: EM algorithm for one-shot device testing with competing risks under Weibull distribution. *IEEE Trans. Reliab.* **65**(2), 973–991 (2016)
14. Balakrishnan, N., Castilla, E., Martin, N., Pardo, L.: Robust estimators and test statistics for one-shot device testing under the exponential distribution. *IEEE Trans. Inform. Theory* **65**(5), 3080–3096 (2019)
15. Balakrishnan, N., Castilla, E., Martin, N., Pardo, L.: Robust estimators for one-shot device testing data under gamma lifetime model with an application to a tumor toxicological data. *Metrika* (2019). <https://doi.org/10.1007/s00184-019-00718-5>
16. Cox, D.R.: Regression models and life tables. *J. R. Stat. Soc. Ser. B* **34**(2), 187–220 (1972)
17. Cox, D.R.: *Analysis of Survival Data*. Routledge, Milton Park (2018)
18. Davidson, R.: Reliable inference for the Gini index. *J. Econ.* **150**(1), 30–40 (2009)
19. Fan, T.H., Balakrishnan, N., Chang, C.C.: The Bayesian approach for highly reliable electro-explosive devices using one-shot device testing. *J. Stat. Comput. Simul.* **79**(9), 1143–1154 (2009)
20. Guo, H., Honecker, S., Mettas, A., Ogden, D.: Reliability estimation for one-shot systems with zero component test failures. In: *Proceedings-Annual Reliability and Maintainability Symposium (RAMS)*, pp. 1–7 (2010)
21. Hoyle, M.H.: Transformations: an introduction and a bibliography. *Int. Stat. Rev.* **41**(2), 203–223 (1973)
22. Hwang, C.P., Ke, H.Y.: A reliability analysis technique for quantal-response data. *Reliab. Eng. Syst. Saf.* **41**(3), 239–243 (1993)
23. Ling, M.H.: Optimal design of simple step-stress accelerated life test for one-shot devices under exponential distributions. *Probab. Eng. Inf. Sci.* **33**(1) 121–135 (2019)
24. Ling, M.H., Balakrishnan, N.: Model mis-specification analyses of Weibull and gamma models for one-shot device testing data. *IEEE Trans. Reliab.* **66**(3), 641–650 (2017)
25. Ling, M.H., So, H.Y., Balakrishnan, N.: Likelihood inference under proportional hazards model for one-shot device testing. *IEEE Trans. Reliab.* **65**(1), 446–458 (2016)
26. Ling, M.H., Chan, P.S., Ng, H.K.T., Balakrishnan, N.: Copula models for one-shot device testing data with correlated failure modes. *Commun. Stat. Theory Methods* **50**(16), 3875–3888 (2021)
27. Ling, M.H., Hu, X.M.: Optimal design of simple step-stress accelerated life tests for one-shot devices under Weibull distributions. *Reliab. Eng. Syst. Saf.* **193**, 106630 (2020)

28. Louis, T.A.: Finding the observed information matrix when using the EM algorithm. *J. R. Stat. Soc. Ser. B* **44**(2), 226–233 (1982)
29. Madi, M.T.: Multiple step-stress accelerated life test: the tampered failure rate model. *Commun. Stat. Theory Methods* **22**(9), 2631–2639 (1993)
30. McLachlan, G., Krishnan, T.: *The EM Algorithm and Extensions*, 2nd. edn. John Wiley & Sons, Hoboken (2007)
31. Meeker, W.Q., Escobar, L.A.: *Statistical Methods for Reliability Data*. John Wiley & Sons, Hoboken (2014)
32. Morris, M.D.: A sequential experimental design for estimating a scale parameter from quantal life testing data. *Technometrics* **29**(2), 173–181 (1987)
33. Musallam, M., Yin, C., Bailey, C., Johnson, C.M.: Application of coupled electro-thermal and physics-of-failure-based analysis to the design of accelerated life tests for power modules. *Microelectron. Reliab.* **54**(1), 172–181 (2014)
34. Nelsen, R.B.: *An Introduction to Copulas*. Springer Science & Business Media, Berlin (2007)
35. Nelson, W.B.: Accelerated life testing – step-stress models and data analyses. *IEEE Trans. Reliab.* **29**(2), 103–108 (1980)
36. Nelson, W.B.: *Accelerated Testing: Statistical Models, Test Plans, and Data Analysis*. John Wiley & Sons, Hoboken (2009)
37. Pan, C.C., Chu, L.: Reliability assessment for one-shot product with Weibull lifetime components. *Int. J. Qual. Reliab. Manag.* **27**(5), 596–610 (2010)
38. Pham, H. (ed.) *Springer Handbook of Engineering Statistics*. Springer Science & Business Media, Berlin (2006)
39. Shaked, M., Singpurwalla, N.D.: A Bayesian approach for quantile and response probability estimation with applications to reliability. *Ann. Inst. Stat. Math.* **42**(1), 1–19 (1990)
40. Sklar, A.: Fonctions de repartition an dimensions et leurs marges. *Publications de l'Institut de statistique de l'Université de Paris* **8**, 229–231 (1959)
41. Stacy, E.W.: A generalization of the gamma distribution. *Ann. Math. Stat.* **33**(3), 1187–1192 (1962)
42. Virkki, J., Seppälä, T., Frisk, L., Heino, P.: Accelerated testing for failures of tantalum capacitors. *Microelectron. Reliab.* **50**(2), 217–219 (2010)
43. Viveros, R., Balakrishnan, N.: Statistical inference from start-up demonstration test data. *J. Qual. Technol.* **25**(2), 119–130 (1993)
44. Wang, W.D., Kececioglu, D.B.: Fitting the Weibull log-linear model to accelerated life-test data. *IEEE Trans. Reliab.* **49**(2), 217–223 (2000)
45. Warwick, J., Jones, M.C.: Choosing a robustness tuning parameter. *J. Stat. Comput. Simul.* **75**(7), 581–588 (2005)
46. White, H.: Maximum likelihood estimation of misspecified models. *Econometrica* **50**(1), 1–25 (1982)
47. Wilks, S.S.: The large-sample distribution of the likelihood ratio for testing composite hypotheses. *Ann. Math. Stat.* **9**(1), 60–62 (1938)
48. Yun, W.Y., Han, Y.J., Kim, H.W.: Simulation-based inspection policies for a one-shot system in storage over a finite time span. *Commun. Stat. Simul. Comput.* **43**(8), 1979–2003 (2014)



Narayanaswamy Balakrishnan received his B.Sc. and M.Sc. degrees in Statistics from the University of Madras, India, in 1976 and 1978, respectively. He finished his Ph.D. in Statistics from the Indian Institute of Technology, Kanpur, India, in 1981. He is a Distinguished University Professor at McMaster University, Hamilton, ON, Canada. His research interests include distribution theory, ordered data analysis, censoring methodology, reliability, survival analysis, nonparametric inference, and statistical quality control. Prof. Balakrishnan was made a Fellow of the American Statistical Association in 1995, a Fellow of the Institute of Mathematical Statistics in 2006, and a Fellow of the Royal Society of Canada in 2021. He is currently the Editor-in-Chief of *Communications in Statistics*. He has received an honorary doctorate degree from The National University of Athens in 2016 for his valuable contributions to the field of Statistics.



Man Ho Ling received the Ph.D. degree from McMaster University, Hamilton, ON, Canada, in 2012. He is an Associate Professor in the Department of Mathematics and Information Technology at the Education University of Hong Kong, Hong Kong SAR, China. His research interests include reliability and survival analyses, statistical inference, censoring methodology, and statistical computing methods.



Tangent Space Approximation in Geometric Statistics

53

Ted Chang

Contents

53.1	The Spherical Regression and Procrustes Models	1060
53.2	Least Squares Estimation	1061
53.2.1	A Simple Example Derivation	1061
53.3	Differential Manifolds	1062
53.4	Tangent Space Approximations	1063
53.4.1	The Lesson from Taylor Series	1063
53.4.2	The Map Maker’s Dilemma and Its Lessons	1063
53.5	Asymptotic Analysis for Procrustes and Spherical Regressions in $k = 3$ Dimensions	1064
53.5.1	Large Sample Asymptotics for the Procrustes Regression Model in R^3	1064
53.5.2	Large κ Asymptotic Analysis for the Spherical Regression Model on S^2	1065
53.6	The Reconstruction of Tectonic Plate Motions from Magnetic Anomaly Lineations	1066
53.6.1	The Hellinger Criterion to Fit Estimate a Reconstruction	1067
53.6.2	Statistical Analysis of the Hellinger Estimate	1068
53.6.3	Additional Work Using Hellinger Estimates	1069
53.7	Summary	1070
Appendix	1070
	Quaternion Representation of $SO(3)$	1070
	Combining Two Rotations Using Quaternionic Representation	1071
	Combining Two Rotations Using Axis Latitude, Longitude, and Angle of Rotation	1071
	The Map Maker’s Dilemma (Revisited)	1072
References	1072

Abstract

The *Procrustes regression* model provides a statistical framework to assess the errors in image registration (in arbitrary dimensions) from “landmark” data. The same mathematics can be used to determine the errors in calculated motions of rigid bodies in Euclidean space.

Perhaps the scientifically most compelling example of rigid body motion is tectonic plates. Tectonic plates, to a first approximation, move as rigid bodies on the surface of the Earth. The estimation of the past configuration of the tectonic plates, and the *errors in these reconstructions*, is integral to the understanding of the past history of the Earth. Because tectonic plates are restricted to the surface of the Earth, the Procrustes regression model does not apply and the relevant model is called *spherical regression*.

The Procrustes and spherical regression models are mathematically simple and, because of this simplicity, beautiful theorems about the properties of their estimates can be proven. The previous chapter (Chang T, Image registration, rigid bodies, and unknown coordinate systems. In: Pham H (ed) Springer handbook of engineering statistics, Springer-Verlag, London, pp. 571–590, 2006) discusses many of these results. Using the spherical regression model, interesting insights into the properties of tectonic plate reconstructions are discussed in (Chang, J Geophys Res 92(B7):6319–6329, 1987; Chang, Int Stat Rev 61:299–316, 1993). We will not replicate these results here. Rather the focus of this chapter is the underlying mathematics used to establish the results. These problems are intrinsically geometric and the proper use of geometry is important to their understanding. We will explain these points.

This chapter is motivated by a request from a geophysics friend to explain, in as elementary fashion as feasible, the mathematics behind his work. This chapter will not replicate the formal proofs that appear elsewhere (see, for example, Chang, Ann Stat 14(3):907–924, 1986, Rivest, Ann Stat 17(1):307–317, 1989, or Chang and Ko,

T. Chang (✉)
 Department of Statistics, University of Virginia, Charlottesville, VA,
 USA
 e-mail: Tcc8v@virginia.edu

Ann Stat 23(5):1823–1847, 1995). This chapter is aimed at the scientist who wants to understand on a heuristic level why the results are true, without reading mathematically complete proofs!

The type of data that is actually used to estimate plate reconstructions (experimentally determined “marine magnetic anomaly lineation” locations) is not of the form that is modeled in the spherical regression model. We discuss in Sect. 53.6 the type of data that is actually collected and how to analyze it using the mathematical principles we will discuss here. For the nongeophysicist, this section can be used as an example of the use of the mathematical principles of this chapter in a different and complex data setting.

Furthermore, although the rigid plate hypothesis is a simplification, one must first understand what types of reconstruction errors are consistent with rigid plates before deciding if the errors one is observing are in fact evidence of nonrigidity. Section 53.6 discusses some work of this type.

Essentially, this chapter is a case study in the use of tangent space approximations and some elementary ideas from differential geometry. Other authors have used similar geometrical approaches. For example, Rivest (J Biomech 38:1604–1611, 2005) and Oualkacha and Rivest (Biometrika 99:585–598, 2012) developed statistical methods for human motion studies derived from sensors placed on the human body as it moves. Chang and Rivest (Ann Stat 29(3):784–814, 2001) extended to Stiefel manifolds the work outlined here and in (Chang T, Image registration, rigid bodies, and unknown coordinate systems. In: Pham H (ed) Springer handbook of engineering statistics, Springer-Verlag, London, pp. 571–590, 2006) and used the results to reanalyze a data set on vector cardiograms. Patrangenaru has numerous papers (e.g., Mardia and Patrangenaru (Ann Stat 33(4):1666–1699, 2005)) developing statistical methods for comparing images when projective, rather than rigid, transformations are allowed. Indeed, the author believes that the engineering disciplines have multiple problems of a geometric nature and hopes that the approaches used here can be helpful in studying them.

This chapter has been written so that it can be read independently of the preceding chapter (Chang T, Image registration, rigid bodies, and unknown coordinate systems. In: Pham H (ed) Springer handbook of engineering statistics, Springer-Verlag, London, pp. 571–590, 2006).

Keywords

Procrustes regression · Spherical regression · Rigid body motion · Tectonic plate reconstruction · Geometric statistics · Tangent space approximation · Quaternions · Cayley-Klein parameters

53.1 The Spherical Regression and Procrustes Models

We will use R^k to denote k -dimensional Euclidean space. A vector $\mathbf{x} \in R^k$ will be thought of as a $k \times 1$ column vector. In particular, $\mathbf{x}^T \mathbf{y}$ is the dot product of \mathbf{x} and \mathbf{y} (using matrix notation) and the length $\|\mathbf{x}\|$ of \mathbf{x} is $\sqrt{\mathbf{x}^T \mathbf{x}}$.

Thus, the unit sphere, of dimension $k - 1$, in R^k will be

$$S^{k-1} = \Omega_k = \{\mathbf{x} \in R^k \mid \|\mathbf{x}\|^2 = \mathbf{x}^T \mathbf{x} = 1\}.$$

(The notation S^{k-1} emphasizes the dimension of the sphere, whereas the notation Ω_k emphasizes that each point in Ω_k is represented as a k -component vector.)

Although the spherical regression model, and its resulting theorems, can be developed in arbitrary dimensions k , $k = 3$ is the case of usual interest so that S^2 represents the Earth.

The spherical regression model posits fixed points $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n \in S^{k-1}$, homologous (“my nose is your nose”) random, and independent points $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \in S^{k-1}$, so that

$$\mathbf{v}_i = \mathbf{A} \mathbf{u}_i + \text{error}. \quad (53.1)$$

\mathbf{A} represents a $k \times k$ matrix that satisfies the *orthogonality* condition

$$\mathbf{A}^T \mathbf{A} = \mathbf{I}_k = \mathbf{A} \mathbf{A}^T, \quad (53.2)$$

where \mathbf{I}_k represents the $k \times k$ identity matrix. This condition insures that each point $\mathbf{A} \mathbf{u}_i \in S^{k-1}$. It also insures a rigidity condition that

$$\|\mathbf{A} \mathbf{u}_i - \mathbf{A} \mathbf{u}_j\| = \|\mathbf{u}_i - \mathbf{u}_j\|. \quad (53.3)$$

Here the rigidity condition is described in terms of Euclidean space distance, but that is equivalent to a rigidity condition in great circle distance for points that lie on the sphere. The collection of such matrices \mathbf{A} is denoted $O(k)$ and referred to as the *orthogonal* matrices.

In the tectonic plate context, the \mathbf{u}_i are present day landmark points on one plate, \mathbf{v}_i are homologous landmark points on the opposing plate, and \mathbf{A}^T reconstructs the past position of the V-plate to the U-plate in a coordinate system fixed to the U-plate.

Usually it is impossible for a rigid body to be “pushed through the center of the Earth” and so there is an additional *orientation preserving* condition that the determinant of \mathbf{A} be 1. The collection of such matrices is denoted $SO(k)$ and referred to as the *special orthogonal* matrices. In particular, $SO(3)$ consists of the 3×3 rotation matrices. Each rotation in $SO(3)$ is defined by an axis, which can be an arbitrary unit vector in S^2 , and an angle of rotation.

The Procrustes regression model (without scale change) posits fixed points $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n \in R^k$ and random points $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \in R^k$, so that each \mathbf{v}_i has a multivariate normal distribution with mean vector $\mathbf{A}\mathbf{u}_i + \mathbf{b}$ and variance covariance matrix $\sigma^2 \mathbf{I}_k$. \mathbf{A} , as before, represents a rotation matrix, and \mathbf{b} a translation vector. This model might apply, for example, if the $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n$ and $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$ are homologous landmarks on two images, and the determination of \mathbf{A} and \mathbf{b} is commonly referred to as *image registration*.

For reasons of convenience in Sect. 53.5.1, we reparameterize the Procrustes model by letting $\boldsymbol{\alpha} = \mathbf{A}\bar{\mathbf{u}} + \mathbf{b}$. Then each \mathbf{v}_i has mean vector $\mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) + \boldsymbol{\alpha}$.

Speaking with unnecessary precision, the “error” in (53.1) is so that the density of each \mathbf{v} is, with respect to surface measure on S^{k-1} , proportional to $e^{\kappa \mathbf{v}^T \mathbf{A}\mathbf{u}}$. This is the “Fisher-von Mises” distribution $F(\mathbf{A}\mathbf{u}, \kappa)$. κ is the *concentration* parameter and $\mathbf{A}\mathbf{u}$ the *modal vector*.

More heuristically, we can write \mathbf{v} in its normal and tangential components

$$\mathbf{v} = (\mathbf{v}^T \mathbf{A}\mathbf{u}) \mathbf{A}\mathbf{u} + (\mathbf{v} - (\mathbf{v}^T \mathbf{A}\mathbf{u}) \mathbf{A}\mathbf{u}). \quad (53.4)$$

The tangential component $\mathbf{v} - (\mathbf{v}^T \mathbf{A}\mathbf{u}) \mathbf{A}\mathbf{u}$ is perpendicular to $\mathbf{A}\mathbf{u}$ and hence is tangent to S^{k-1} at $\mathbf{A}\mathbf{u}$. We will use $(\mathbf{A}\mathbf{u})^\perp$ to denote the collection of vectors perpendicular to $\mathbf{A}\mathbf{u}$. If the error is small, which occurs for large κ , $(\mathbf{v}^T \mathbf{A}\mathbf{u}) \mathbf{A}\mathbf{u}$ will be close to $\mathbf{A}\mathbf{u}$ and the tangential component $\mathbf{v} - (\mathbf{v}^T \mathbf{A}\mathbf{u}) \mathbf{A}\mathbf{u}$ will be closely approximated by a rotationally symmetric $k - 1$ dimensional multivariate normal distribution $N(\mathbf{0}, \kappa^{-1} \mathbf{I})$ with mean vector $\mathbf{0}$ and variance covariance matrix $\kappa^{-1} \mathbf{I}$. Here \mathbf{I} represents an identity matrix in $(\mathbf{A}\mathbf{u})^\perp$. Mathematically, it can be written as a $k \times k$ matrix as $\mathbf{I} = \mathbf{I}_k - (\mathbf{A}\mathbf{u})(\mathbf{A}\mathbf{u})^T$.

It is essential in what follows to understand that we are thinking of the hopefully small (that is large κ) error in \mathbf{v} from its “true” value $\mathbf{A}\mathbf{u}$ as a small vector (close to $\mathbf{0}$) in the tangent space $(\mathbf{A}\mathbf{u})^\perp$. Tangent space approximation is the essential mathematical tool being used in spherical regression theory. Thus, in what follows, we will let $\boldsymbol{\varepsilon} = \mathbf{v} - (\mathbf{v}^T \mathbf{A}\mathbf{u}) \mathbf{A}\mathbf{u}$ and rewrite (53.4) as

$$\mathbf{v} = \sqrt{1 - \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}} \mathbf{A}\mathbf{u} + \boldsymbol{\varepsilon} \approx \mathbf{A}\mathbf{u} + \boldsymbol{\varepsilon} \quad (53.5)$$

where the error vector $\boldsymbol{\varepsilon}$ is normally distributed, in the tangent space $(\mathbf{A}\mathbf{u})^\perp$, with variance covariance matrix $\kappa^{-1} \mathbf{I}$. If κ is large, then $\boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}$ is small compared to $\boldsymbol{\varepsilon}$ and hence the approximation in (53.5). Indeed $\boldsymbol{\varepsilon}$ has order $\kappa^{-1/2}$ (denoted $O_p(\kappa^{-1/2})$ – the subscript p stands for *probability*), whereas $\boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}$ is $O_p(\kappa^{-1})$ so that as $\kappa \rightarrow \infty$, it is asymptotically negligible.

We note that the errors in magnetic anomaly lineation data are small compared to the radius of the Earth, so large κ approximation is appropriate.

53.2 Least Squares Estimation

In the spherical regression model, the least squares estimate $\hat{\mathbf{A}}$ of \mathbf{A} minimizes the objective function

$$\rho_2(\mathbf{A}) = \sum \| \mathbf{v}_i - \mathbf{A}\mathbf{u}_i \|^2 \quad (53.6)$$

and, in general, the L_p estimate minimizes $\rho_p(\mathbf{A}) = \sum \| \mathbf{v}_i - \mathbf{A}\mathbf{u}_i \|^p$. The least squares estimate is computationally easier, and indeed, there is an explicit formula for it in terms of the singular value decomposition of the 3×3 matrix $\sum \mathbf{u}_i \mathbf{v}_i^T$, see for example [1]. That being said, the L_1 estimate can be used if outliers are a potential issue, although numerical minimization is required.

In the Procrustes regression model, the least squares estimate $\hat{\mathbf{A}}$ of \mathbf{A} and $\hat{\boldsymbol{\alpha}}$ of $\boldsymbol{\alpha}$ minimize the objective function

$$\rho_2(\mathbf{A}, \boldsymbol{\alpha}) = \sum \| \mathbf{v}_i - \mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) - \boldsymbol{\alpha} \|^2 \quad (53.7)$$

Section 53.5 derives some asymptotic distributions of these least squares estimators. The first edition of this chapter [1] has more general results that include, for example, L_p estimators.

53.2.1 A Simple Example Derivation

Before deriving in Sect. 53.5, the distribution of $\hat{\mathbf{A}}$ (for the spherical regression problem) or for $\hat{\mathbf{A}}$ and $\hat{\boldsymbol{\alpha}}$ (for unscaled Procrustes regression), consider the following simpler problem of *location* estimation. Suppose $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \in S^{k-1}$ are independently distributed $F(\boldsymbol{\mu}, \kappa)$ where $\boldsymbol{\mu} \in S^{k-1}$ is the modal (location) vector and κ is large. We want to figure out the distribution of the estimate $\hat{\boldsymbol{\mu}}$ that minimizes

$$\rho_2(\boldsymbol{\mu}) = \sum \| \mathbf{v}_i - \boldsymbol{\mu} \|^2. \quad (53.8)$$

Write, in a manner similar to (53.5),

$$\mathbf{v}_i \approx \boldsymbol{\mu} + \boldsymbol{\varepsilon}_i \quad (53.9)$$

$$\hat{\boldsymbol{\mu}} \approx \boldsymbol{\mu} + \mathbf{u} \quad (53.10)$$

where $\boldsymbol{\varepsilon}_i, \mathbf{u} \in \boldsymbol{\mu}^\perp$, that is, they are tangent to S^{k-1} at $\boldsymbol{\mu}$. Using the approximations (53.9) and (53.10), (53.8) becomes

$$\rho_2(\hat{\boldsymbol{\mu}}) \approx \sum \| \boldsymbol{\mu} + \boldsymbol{\varepsilon}_i - (\boldsymbol{\mu} + \mathbf{u}) \|^2 = \sum \| \boldsymbol{\varepsilon}_i - \mathbf{u} \|^2 \quad (53.11)$$

and \mathbf{u} minimizes (53.11). Since each $\boldsymbol{\varepsilon}_i$ is approximately distributed $N(\mathbf{0}, \kappa^{-1}\mathbf{I})$, $\mathbf{u} = n^{-1} \sum \boldsymbol{\varepsilon}_i$, and is approximately distributed $N(\mathbf{0}, (n\kappa)^{-1}\mathbf{I})$. Note that these distributional statements occur in the tangent space to S^{k-1} at $\boldsymbol{\mu}$, so that $\mathbf{I} = \mathbf{I}_k - \boldsymbol{\mu}\boldsymbol{\mu}^T$. Also note that we are actually computing the distribution of the deviation \mathbf{u} of $\hat{\boldsymbol{\mu}}$ from $\boldsymbol{\mu}$. Using standard statistical reasoning, this latter distribution is sufficient for both hypothesis tests and confidence regions on the parameter $\boldsymbol{\mu}$.

And, again, these are large κ asymptotic approximations. Technically they hold, in the limit, as $\kappa \rightarrow \infty$.

We also note that an explicit solution for $\hat{\boldsymbol{\mu}}$ is $\hat{\boldsymbol{\mu}} = \sum \mathbf{v}_i / \|\sum \mathbf{v}_i\|$ although this explicit solution is not used in the above.

53.3 Differential Manifolds

In the derivation of Sect. 53.2.1, S^{k-1} appears in two roles: It is both the data space, containing the data points $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$, and the parameter space, containing the parameter $\boldsymbol{\mu}$ to be estimated. In both contexts, tangent space approximations are used: Using (53.9), the $\boldsymbol{\varepsilon}_i$ lies in the tangent space to the data space and in (53.10), \mathbf{u} lies in the tangent space of the parameter space.

In the spherical regression model, S^{k-1} is the data space, but $\text{SO}(k)$ is the parameter space. In the Procrustes model, R^k is the data space and the parameter space is $\text{SO}(k) \times R^k = \{(\mathbf{A}, \mathbf{b})\}$, where \mathbf{A} represents a rotation matrix and \mathbf{b} a translation vector. Thus, besides tangent space approximation to S^{k-1} , we need to understand how to make tangent space approximations to $\text{SO}(k)$. Tangent space approximations to $\text{SO}(k) \times R^k$ are a simple extension.

In addition, using (53.5) and reintroducing subscripts into it, each $\boldsymbol{\varepsilon}_i$ lies in the tangent space $(\mathbf{A}\mathbf{u}_i)^\perp$. So, we will need, for the data space at least, multiple tangent space approximations, one for each data point \mathbf{v}_i . This latest complication is intrinsic to regression problems.

Each element $\mathbf{A} \in \text{SO}(k)$ is a $k \times k$ matrix and has k^2 elements so we can easily represent $\text{SO}(k) \subseteq R^{k^2}$. In the mathematical lingo, S^{k-1} and $\text{SO}(k)$ are “embedded manifolds” in R^k and R^{k^2} , respectively. Differential geometry is the mathematical discipline that studies manifolds and the notion of tangent space is one of that discipline’s most fundamental constructs. Because our manifolds are naturally embedded in ambient Euclidean spaces, we can introduce the ideas we need without using the usual mathematical definitions in their most general and precise formulation.

In fact, in the history of differential geometry, the most important non-Euclidean geometry is the sphere S^2 . The sum of the angles in a spherical triangle (that is a triangle bounded by three great circle segments) always exceeds 180° and parallel lines do not exist. More relevantly, for our purposes, the sphere cannot be represented as a planar map without

tearing it somewhere. But small patches of the sphere can be represented using a map and we do so every day. In other words, the fact that the sphere closes in on itself is a global feature; locally we can think of it as a two dimensional plane with two cardinal directions (EW and NS).

A manifold of dimension p is, roughly speaking, a topological object that can locally be represented as a region of R^p .

For example, a polar projection centered at $\mathbf{u} \in S^{k-1}$ maps $\mathbf{h} \in \mathbf{u}^\perp$

$$\mathbf{h} \rightarrow \cos(\|\mathbf{h}\|)\mathbf{u} + \sin(\|\mathbf{h}\|)\|\mathbf{h}\|^{-1}\mathbf{h} \tag{53.12}$$

and this represents S^{k-1} locally at $\mathbf{u} \in S^{k-1}$ as the Euclidean space $\mathbf{u}^\perp \approx R^{k-1}$. (Note that for spaces, the symbol “ \approx ” in $\mathbf{u}^\perp \approx R^{k-1}$ is meant to convey that \mathbf{u}^\perp is a version of Euclidean $(k - 1)$ -dimensional space R^{k-1} , whereas for points, as in the Equations (53.5), (53.9), (53.10), and (53.11), it denotes approximation.) We say that S^{k-1} is a $(k - 1)$ -dimensional manifold, and it is embedded in R^k .

The matrix exponential function takes the skew symmetric $k \times k$ matrices

$$\text{Skew}(k) = \{\mathbf{H} | \mathbf{H} + \mathbf{H}^T = 0\}$$

onto $\text{SO}(k)$. It is defined by

$$\exp(\mathbf{H}) = \sum_{r=0}^{\infty} \frac{\mathbf{H}^r}{r!} \tag{53.13}$$

where \mathbf{H} is a $k \times k$ skew symmetric matrix. Notice that $\text{Skew}(k) \approx R^{k(k-1)/2}$.

For $k = 3$, and the 3-vector $\mathbf{t} = [t_1 \ t_2 \ t_3]^T$, let

$$\Phi(\mathbf{t}) = \exp\left(\begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}\right) = \exp \mathbf{M}(\mathbf{t}) \tag{53.14}$$

where $\mathbf{M}(\mathbf{t})$ is defined by equation (53.14).

It can be shown that for the 3-vector \mathbf{t} , $\Phi(\mathbf{t})$ represents right hand rule rotation of $\|\mathbf{t}\|$ radians around the axis $\|\mathbf{t}\|^{-1}\mathbf{t}$. Thus, if \mathbf{t} is a small vector, $\Phi(\mathbf{t})$ will be a small rotation, that is, one close to \mathbf{I}_3 . Also note that if \mathbf{u} is of unit length $\Phi(2\pi\mathbf{u}) = \mathbf{I}_3$, so that Φ , and more generally $\exp(\mathbf{H})$, cannot globally represent $\text{SO}(k)$ as $\text{Skew}(k)$.

Nevertheless we can use (53.13) to define a manifold structure for $\text{SO}(k)$. For each $\mathbf{A} \in \text{SO}(k)$, let $\mathbf{A}\text{Skew}(k) = \{\mathbf{A}\mathbf{H} | \mathbf{H} \in \text{Skew}(k)\} \approx R^{k(k-1)/2}$.

$$\mathbf{A}\mathbf{H} \rightarrow \mathbf{A} \exp(\mathbf{H}) \tag{53.15}$$

represents the special orthogonal matrices close to \mathbf{A} as a portion of $\mathbf{A}\text{Skew}(k) \approx R^{k(k-1)/2}$. Thus, we can say that $\text{SO}(k)$ is a $k(k - 1)/2$ -dimensional manifold, and it is embedded in R^{k^2} .

53.4 Tangent Space Approximations

Let \mathcal{M} be a p -dimensional manifold embedded in some ambient Euclidean space R^N . If $c(t)$ is a curve in \mathcal{M} , that is, a function $R^1 \rightarrow \mathcal{M}$, it is a fortiori a curve in R^N . Thus, we can take derivatives $c'(t)$ and $c'(t)$ will be a vector in R^N . The vector $c'(0)$ is said to be a *tangent vector* to \mathcal{M} at $c(0) \in \mathcal{M}$. For a specific point $\mathbf{u} \in \mathcal{M}$, the *tangent space* to \mathcal{M} is the collection of all $c'(0)$ for curves with $c(0) = \mathbf{u}$.

For example, if $c(t)$ is a curve in $S^{k-1} \subseteq R^k$, we have for all t

$$c(t)^T c(t) = 1. \quad (53.16)$$

Differentiating (53.16) yields

$$0 = c'(t)^T c(t) + c(t)^T c'(t) = 2c'(t)^T c(t). \quad (53.17)$$

Specializing (53.17), to $t = 0$, we see that $c'(0) \in c(0)^\perp$. Conversely for $\mathbf{u} \in S^{k-1}$ and $\mathbf{h} \in \mathbf{u}^\perp$

$$c(t) = \cos(t \|\mathbf{h}\|) \mathbf{u} + \sin(t \|\mathbf{h}\|) \|\mathbf{h}\|^{-1} \mathbf{h}$$

is a curve in S^{k-1} with $c(0) = \mathbf{u}$ and $c'(0) = \mathbf{h}$.

In other words, the tangent space to S^{k-1} at \mathbf{u} is exactly \mathbf{u}^\perp .

Similarly, if $c(t)$ is a curve in $\text{SO}(k)$, and using (53.2)

$$c(t)^T c(t) = \mathbf{I}_k.$$

Hence,

$$\mathbf{0} = c(0)^T c'(0) + c'(0)^T c(0). \quad (53.18)$$

Let $\mathbf{A} = c(0)$ and $\mathbf{H} = \mathbf{A}^T c'(0)$. Rewriting (53.18)

$$\mathbf{0} = \mathbf{H} + \mathbf{H}^T$$

or $\mathbf{H} \in \text{Skew}(k)$ and $c'(0) = \mathbf{A}\mathbf{H} \in \mathbf{A}\text{Skew}(k)$. Conversely, if $\mathbf{H} \in \text{Skew}(k)$, (53.15) implies that $c(t) = \mathbf{A} \exp(t\mathbf{H})$ is a curve in $\text{SO}(k)$ with $c(0) = \mathbf{A}$ and $c'(0) = \mathbf{A}\mathbf{H}$.

In other words, the tangent space to $\text{SO}(k)$ at $\mathbf{A} \in \text{SO}(k)$ is exactly the collection of $k \times k$ matrices of the form $\mathbf{A}\text{Skew}(k)$.

53.4.1 The Lesson from Taylor Series

In (53.9) and (53.10), deviations in \mathbf{v}_i and $\hat{\boldsymbol{\mu}}$ from their “true” (without error) value $\boldsymbol{\mu}$ are expressed using tangent vectors $\boldsymbol{\varepsilon}_i$ and \mathbf{u} . The remainder of the proof focuses on the distributions of these tangent vectors.

The linear term of Taylor series

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \text{Remainder}$$

best approximates $f(x) - f(x_0)$, that is, the remainder is small, when x is close to x_0 . This means that when we can guarantee that \mathbf{v}_i and $\hat{\boldsymbol{\mu}}$ are close to $\boldsymbol{\mu}$, we should focus on their deviations from $\boldsymbol{\mu}$, as measured by $\boldsymbol{\varepsilon}_i$ and \mathbf{u} and not upon \mathbf{v}_i and $\hat{\boldsymbol{\mu}}$ directly.

Indeed, in neglecting terms of order $\|\boldsymbol{\varepsilon}_i\|^2$ and $\|\mathbf{u}\|^2$ when passing from $\|\mathbf{v}_i - \boldsymbol{\mu}\|$ in (53.11) to $\|\boldsymbol{\varepsilon}_i - \mathbf{u}\|$ in (53.6), we have applied Taylor linearization in both the data space and the parameter space. This is justifiable because as $\kappa \rightarrow \infty$, $\mathbf{v}_i \rightarrow \boldsymbol{\mu}$ and $\hat{\boldsymbol{\mu}} \rightarrow \boldsymbol{\mu}$, so that, in the limit, $\|\boldsymbol{\varepsilon}_i\|^2$ and $\|\mathbf{u}\|^2$ are asymptotically trivial.

In the plate tectonic context, the errors in tectonic plate data are on the order of 10 km and hence trivial relative to the radius of the Earth. This justifies a “local flat Earth approximation,” that is, Taylor linearization is justified in the data space.

Taylor linearization in the parameter space requires showing that $\hat{\boldsymbol{\mu}} \rightarrow \boldsymbol{\mu}$. This *large κ consistency* is usually very easy to establish unless the underlying statistical model has been badly specified.

Although statistical theory is rife with asymptotic approximations, most of them are *large sample* approximations; they are valid in the limit as the sample size $n \rightarrow \infty$. In this case, one only has consistency in the parameter space, and Taylor linearization in the data space is not available (one generally uses Central Limit type theorems instead). For this reason, the large sample results for spherical regression (see for example [4]) are not as clean as the corresponding large κ results (as in Rivest [5]).

53.4.2 The Map Maker’s Dilemma and Its Lessons

The approximations in (53.9) and (53.10) are essentially maps of tangent vectors $\boldsymbol{\varepsilon}_i$ and \mathbf{u} at $\boldsymbol{\mu}$ into points \mathbf{v}_i and $\hat{\boldsymbol{\mu}}$ on S^{k-1} .

An important property of non-Euclidean manifolds, such as S^{k-1} and $\text{SO}(k)$, is that any representation (map) of them as portion of Euclidean space distorts distances at least to some degree. Thus, any map of any portion of the Earth distorts distances somewhere. For example, the standard Mercator projection of the Earth is highly distorting in polar regions. Thus, when making a map of a region on the Earth, a map maker must choose a projection that makes the distortions as small as possible in the region of primary interest. So a map of Antarctica is not made using a standard Mercator projection, but rather using a polar projection (53.12) centered at the South pole.

A body is rigid when its internal distances are preserved as it moves. Thus, the problem of rigid body motion (or moving

coordinate systems) intrinsically involves distances, and any maps we use of S^{k-1} should, as best possible, preserve distances. Because v_i and $\hat{\boldsymbol{\mu}}$ are close to $\boldsymbol{\mu}$, we are called to use a polar projection of $\boldsymbol{\mu}^\perp$ onto S^{k-1} , that is, a polar projection centered at $\boldsymbol{\mu}$.

The polar projection (53.12) centered at $\mathbf{u} \in S^{k-1}$ takes $\mathbf{h} \in \mathbf{u}^\perp$ to

$$\mathbf{h} \rightarrow \cos(\|\mathbf{h}\|)\mathbf{u} + \sin(\|\mathbf{h}\|)\|\mathbf{h}\|^{-1}\mathbf{h} = \mathbf{u} + \mathbf{h} + \mathcal{O}(\|\mathbf{h}\|^2)$$

so that using (53.9) and (53.10) are asymptotically equivalent to using a polar projection centered at $\boldsymbol{\mu}$.

It can be shown that the map (53.15) of the tangent space $\mathbf{ASkew}(k)$ to $\text{SO}(k)$ at $\mathbf{A} \in \text{SO}(k)$ has similar distance properties as polar projection on S^{k-1} . (This requires somewhat more understanding of the geometry of $\text{SO}(k)$, but for $k = 3$, a more elementary explanation can be given, see the Appendix sections “Quaternion Representation of $\text{SO}(3)$ ” and “Combining Two Rotations Using Quaternionic Representation”.)

[3, 11] contain a striking example in $k = 3$ of the mess that can be made if statistical approaches are designed without proper use of geometry. In these chapters, the same basic Taylor linearization tool was used in both the messy and clean results. The difference between the messy and clean results is that the clean results (Eq. (53.35) below) were derived by linearizing $\text{SO}(3)$ using tangent space approximation and equation (53.15), whereas the messy results used a different commonly used mapping (axis latitude, axis longitude, angle of rotation) of $\text{SO}(3)$. This example is developed further in the Appendix.

Note that the map (53.15) relies on the group nature of $\text{SO}(k)$. In the terminology of statistical group models, using (53.15) yields “invariant” results and [11] also explains the importance of invariant inference in the geophysical context.

53.5 Asymptotic Analysis for Procrustes and Spherical Regressions in $k = 3$ Dimensions

In this section, we illustrate the use of Taylor linearization for least squares estimation in both the spherical regression and Procrustes models. The results we derive are special cases of those given in Theorem 31.4.4 (for Procrustes models) and Sect. 31.4.9 (for spherical regression models) of [1].

Use of Taylor linearization requires proof of *consistency*, that is that in the limit, the parameter estimates approach their true values. Consistency insures that in the limit, the nonlinear terms will be negligible relative to the linear terms. For the Procrustes model, we need that $\hat{\mathbf{A}} \rightarrow \mathbf{A}$ and $\hat{\boldsymbol{\alpha}} \rightarrow \boldsymbol{\alpha}$. We will perform our asymptotics as the sample size $n \rightarrow \infty$. Although we will not prove consistency, it can be shown that

$\hat{\mathbf{A}}$ and $\hat{\boldsymbol{\alpha}}$ are consistent except for the unreasonable geometry that $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n$ are all multiples of a single vector.

As discussed in Sect. 53.4.1, large sample asymptotics essentially linearizes the parameter space. For spherical regression, we will use large κ asymptotics to linearize both the data and the parameter space. Large sample asymptotics are available for spherical regressions (see [1]), but both their derivations and statements are more difficult (see [4]).

The parameter space for spherical regression on S^{k-1} is $\text{SO}(k)$. For Procrustes regression on R^k , it is $\text{SO}(k) \times R^k$. $\text{SO}(k)$ has dimension $k(k - 1)/2$ so to simplify matters we will restrict ourselves to the special case $k = 3$. This case is also the one of greatest physical interest.

53.5.1 Large Sample Asymptotics for the Procrustes Regression Model in R^3

In the example proof of Sect. 53.2.1, both the data space and the sample space were linearized using equations (53.9) and (53.10). For Procrustes regression, the data space is already linear and satisfies the model

$$v_i = \mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) + \boldsymbol{\alpha} + \boldsymbol{\varepsilon}_i \tag{53.19}$$

Let $(\hat{\mathbf{A}}, \hat{\boldsymbol{\alpha}})$ minimize (53.7) and write

$$\hat{\mathbf{A}} = \mathbf{A} \exp \mathbf{H} \approx \mathbf{A}(\mathbf{I} + \mathbf{H}) \quad \text{and} \quad \hat{\boldsymbol{\alpha}} = \boldsymbol{\alpha} + \mathbf{h}_\alpha \tag{53.20}$$

where in (53.20) the asymptotically trivial terms of order \mathbf{H}^2 and higher have been ignored

Equations (53.19) and (53.20) play an analogous role in the derivation of this section that (53.9) and (53.10) played in the derivation of Sect. 53.2.1: They are the expression of the deviations of the data v_i and the parameter estimates $(\hat{\mathbf{A}}, \hat{\boldsymbol{\alpha}})$ from their “true” values $\mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) + \boldsymbol{\alpha}$ and $(\mathbf{A}, \boldsymbol{\alpha})$. The data space R^k is its own tangent space and we are thinking of $\boldsymbol{\varepsilon}_i$ as a tangent vector to R^k at $\mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) + \boldsymbol{\alpha}$. $(\mathbf{A}\mathbf{H}, \mathbf{h}_\alpha)$ is a tangent vector to the parameter space $\text{SO}(k) \times R^k$ at $(\mathbf{A}, \boldsymbol{\alpha})$.

The distribution of \mathbf{H} will be in $k(k - 1)/2$ dimensions and the entries of \mathbf{H} are naturally double subscripted. This makes the derivation somewhat hard to follow, although its fundamental idea is almost as simple as the derivation in Sect. 53.2.1. To alleviate this problem partially, we will specialize to the case of physical interest $k = 3$.

Thus, as in equation (53.14), we write \mathbf{H} in the form $\mathbf{H} = \exp \mathbf{M}(\mathbf{h})$ and $\hat{\mathbf{A}} = \mathbf{A}\Phi(\mathbf{h})$ where $\mathbf{h} = [h_1 \ h_2 \ h_3]^T$ is a 3×1 column vector. We seek the asymptotic distribution of $(\mathbf{h}, \mathbf{h}_\alpha)$.

Recall that $\hat{\mathbf{A}}$ minimizes the objective function $\rho_2(\mathbf{A}, \boldsymbol{\alpha})$ of equation (53.7). Substituting (53.19) and (53.20) into (53.7), we arrive at

$$\begin{aligned}
\rho_2(\hat{\mathbf{A}}, \hat{\boldsymbol{\alpha}}) &\approx \sum \|\mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) + \boldsymbol{\alpha} + \boldsymbol{\varepsilon}_i \\
&\quad - \mathbf{A}(\mathbf{I} + \mathbf{H})(\mathbf{u}_i - \bar{\mathbf{u}}) - \boldsymbol{\alpha} - \mathbf{h}_\alpha\|^2 \\
&= \sum \|\boldsymbol{\varepsilon}_i - \mathbf{A}\mathbf{H}(\mathbf{u}_i - \bar{\mathbf{u}}) - \mathbf{h}_\alpha\|^2 \\
&= \sum \|\mathbf{A}^T \boldsymbol{\varepsilon}_i - \mathbf{H}(\mathbf{u}_i - \bar{\mathbf{u}}) - \mathbf{A}^T \mathbf{h}_\alpha\|^2
\end{aligned} \tag{53.21}$$

where the last equality follows from equations (53.2) and (53.3). Equation (53.21) plays the same role in the instant derivation as equation (53.11) plays in the derivation of Sect. 53.2.1.

Routine calculations show $\mathbf{H}\mathbf{t} = \mathbf{M}(\mathbf{h})\mathbf{t} = -\mathbf{M}(\mathbf{t})\mathbf{h} = -\mathbf{t} \times \mathbf{h}$, for any vector \mathbf{t} , where $\mathbf{t} \times \mathbf{h}$ denotes the vector cross product. Let $\mathbf{y}_i = \mathbf{A}^T \boldsymbol{\varepsilon}_i$. Since $\boldsymbol{\varepsilon}_i$ is distributed $N(\mathbf{0}, \sigma^2 \mathbf{I}_3)$ and using (53.2), \mathbf{y}_i is also distributed $N(\mathbf{0}, \sigma^2 \mathbf{I}_3)$. In other words, if the $3n \times 1$ vector \mathbf{y} is obtained by stacking $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n$, it satisfies the assumptions of the regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

$\boldsymbol{\epsilon}$ is distributed $N(\mathbf{0}, \sigma^2 \mathbf{I}_{3n})$.

Here \mathbf{X} is the $3n \times 6$ matrix obtained by stacking vertically the 3×6 matrices $\mathbf{X}_i = \begin{bmatrix} -\mathbf{M}(\mathbf{u}_i - \bar{\mathbf{u}}) & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix}$, $i = 1, \dots, n$. $\boldsymbol{\beta}$ is a 6×1 vector whose “true” value is $\mathbf{0}$. The right hand side of (53.21) is the sum squares to be minimized by this regression. The minimum occurs at $\hat{\boldsymbol{\beta}} = (\mathbf{h}, \mathbf{h}_\alpha)$, where $\hat{\mathbf{A}} = \mathbf{A}\Phi(\mathbf{h})$ and $\hat{\boldsymbol{\alpha}} = \boldsymbol{\alpha} + \mathbf{h}_\alpha$.

Now

$$\mathbf{X}^T \mathbf{X} = \begin{bmatrix} \sum \mathbf{M}(\mathbf{u}_i - \bar{\mathbf{u}})^T \mathbf{M}(\mathbf{u}_i - \bar{\mathbf{u}}) & -\sum \mathbf{M}(\mathbf{u}_i - \bar{\mathbf{u}})^T \mathbf{A}^T \\ -\mathbf{A} \sum \mathbf{M}(\mathbf{u}_i - \bar{\mathbf{u}}) & n \mathbf{A} \mathbf{A}^T \end{bmatrix}.$$

For any vector \mathbf{t} , routine algebra shows $\mathbf{M}(\mathbf{t})^T \mathbf{M}(\mathbf{t}) = \|\mathbf{t}\|^2 \mathbf{I}_3 - \mathbf{t}\mathbf{t}^T$ and $\sum \mathbf{M}(\mathbf{u}_i - \bar{\mathbf{u}}) = \mathbf{M}(\sum (\mathbf{u}_i - \bar{\mathbf{u}})) = \mathbf{0}$. Let $\boldsymbol{\Sigma} = n^{-1} \sum (\mathbf{u}_i - \bar{\mathbf{u}})(\mathbf{u}_i - \bar{\mathbf{u}})^T$. Furthermore, $\sum \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2 = \sum \text{Tr}((\mathbf{u}_i - \bar{\mathbf{u}})^T (\mathbf{u}_i - \bar{\mathbf{u}})) = \sum \text{Tr}((\mathbf{u}_i - \bar{\mathbf{u}})(\mathbf{u}_i - \bar{\mathbf{u}})^T) = n \text{Tr}(\boldsymbol{\Sigma})$. Then

$$\mathbf{X}^T \mathbf{X} = n \begin{bmatrix} \text{Tr}(\boldsymbol{\Sigma}) \mathbf{I}_3 - \boldsymbol{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_3 \end{bmatrix}.$$

Since $\hat{\boldsymbol{\beta}}$ has covariance matrix $\sigma^2 (\mathbf{X}^T \mathbf{X})^{-1}$, we have

- \mathbf{h} and \mathbf{h}_α are asymptotically independent.
- \mathbf{h} is asymptotically distributed $N(\mathbf{0}, n^{-1} \sigma^2 (\text{Tr}(\boldsymbol{\Sigma}) \mathbf{I}_3 - \boldsymbol{\Sigma})^{-1})$.
- \mathbf{h}_α is asymptotically normal $N(\mathbf{0}, n^{-1} \sigma^2 \mathbf{I}_3)$.

Section 31.4.6 of [1] has a numerical example using these results for a test on \mathbf{A} .

53.5.2 Large κ Asymptotic Analysis for the Spherical Regression Model on S^2

Large κ asymptotics for the spherical regression model were introduced by Rivest [5] and we follow his approach. Write as in (53.5)

$$\mathbf{v}_i \approx \mathbf{A}\mathbf{u}_i + \boldsymbol{\varepsilon}_i \tag{53.22}$$

where $\boldsymbol{\varepsilon}_i \in (\mathbf{A}\mathbf{u}_i)^\perp$ and as $\kappa \rightarrow \infty$, the distribution of $\boldsymbol{\varepsilon}_i$ approaches $N(\mathbf{0}, \kappa^{-1}(\mathbf{I}_k - (\mathbf{A}\mathbf{u}_i)(\mathbf{A}\mathbf{u}_i)^T))$. The vector $\mathbf{A}\mathbf{u}_i$ is an eigenvector of the matrix $\mathbf{I}_k - (\mathbf{A}\mathbf{u}_i)(\mathbf{A}\mathbf{u}_i)^T$ with eigenvalue 0. All vectors in $(\mathbf{A}\mathbf{u}_i)^\perp$ are eigenvectors of $\mathbf{I}_k - (\mathbf{A}\mathbf{u}_i)(\mathbf{A}\mathbf{u}_i)^T$ with eigenvalue 1. Thus, we have made explicit that each $\boldsymbol{\varepsilon}_i$ is constrained to lie to the tangent space $(\mathbf{A}\mathbf{u}_i)^\perp$. Let

$$\hat{\mathbf{A}} = \mathbf{A} \exp \mathbf{H} \approx \mathbf{A}(\mathbf{I} + \mathbf{H}). \tag{53.23}$$

Again, restricting to $k = 3$, write $\hat{\mathbf{A}} = \mathbf{A}\Phi(\mathbf{h})$ where $\mathbf{h} = [h_1 \ h_2 \ h_3]^T$ is a 3×1 column vector. We seek the asymptotic distribution of \mathbf{h} .

Substituting (53.22) and (53.23) into the objective function $\rho_2(\hat{\mathbf{A}})$ of equation (53.6), we arrive at

$$\begin{aligned}
\rho_2(\hat{\mathbf{A}}) &\approx \sum \|\mathbf{A}\mathbf{u}_i + \boldsymbol{\varepsilon}_i - \mathbf{A}(\mathbf{I} + \mathbf{H})\mathbf{u}_i\|^2 \\
&= \sum \|\mathbf{A}^T \boldsymbol{\varepsilon}_i - \mathbf{H}\mathbf{u}_i\|^2 = \sum \|\mathbf{A}^T \boldsymbol{\varepsilon}_i + \mathbf{M}(\mathbf{u}_i)\mathbf{h}\|^2.
\end{aligned} \tag{53.24}$$

Note that $\mathbf{A}^T \boldsymbol{\varepsilon}_i$ and $-\mathbf{M}(\mathbf{u}_i)\mathbf{h} = -\mathbf{u}_i \times \mathbf{h}$ are three component vectors, but both lie in $\mathbf{u}_i^\perp \approx R^2$. To use standard regression theory, we will reexpress them in terms of a basis of \mathbf{u}_i^\perp . Formally, for each i , pick unit length vectors $\mathbf{v}_i, \mathbf{w}_i$ so that $\mathbf{u}_i, \mathbf{v}_i, \mathbf{w}_i$ form a right hand rule orthonormal basis. In other words, $\mathbf{U}_i = [\mathbf{u}_i \ \mathbf{v}_i \ \mathbf{w}_i] \in SO(3)$ and

$$\mathbf{u}_i^T \mathbf{u}_i = \mathbf{v}_i^T \mathbf{v}_i = \mathbf{w}_i^T \mathbf{w}_i = 1 \tag{53.25}$$

$$\mathbf{u}_i^T \mathbf{v}_i = \mathbf{v}_i^T \mathbf{w}_i = \mathbf{w}_i^T \mathbf{u}_i = 0 \tag{53.26}$$

$$\mathbf{u}_i \mathbf{u}_i^T + \mathbf{v}_i \mathbf{v}_i^T + \mathbf{w}_i \mathbf{w}_i^T = \mathbf{I}_3. \tag{53.27}$$

(53.25) and (53.26) follow from $\mathbf{U}_i^T \mathbf{U}_i = \mathbf{I}_3$ and (53.27) from $\mathbf{U}_i \mathbf{U}_i^T = \mathbf{I}_3$.

Let \mathbf{V}_i be the 3×2 matrix whose columns are \mathbf{v}_i and \mathbf{w}_i and let $\tilde{\boldsymbol{\varepsilon}}_i = \mathbf{V}_i^T \mathbf{A}^T \boldsymbol{\varepsilon}_i$. $\tilde{\boldsymbol{\varepsilon}}_i$ is a 2-dimensional vector and its variance covariance matrix of $\tilde{\boldsymbol{\varepsilon}}_i$ is (for large κ)

$$\kappa^{-1} \mathbf{V}_i^T \mathbf{A}^T (\mathbf{I}_3 - (\mathbf{A}\mathbf{u}_i)(\mathbf{A}\mathbf{u}_i)^T) \mathbf{A} \mathbf{V}_i = \kappa^{-1} \mathbf{I}_2.$$

We have made generous use here of equations (53.25) and (53.26). Furthermore, (53.24) becomes

$$\rho_2(\hat{\mathbf{A}}) \approx \sum \|\tilde{\mathbf{e}}_i + \mathbf{V}_i^T \mathbf{M}(\mathbf{u}_i) \mathbf{h}\|^2.$$

Let \mathbf{X}_i be the 2×3 matrix with rows \mathbf{w}_i^T and $-\mathbf{v}_i^T$. It is well known that for three vectors $\mathbf{u}^T(\mathbf{v} \times \mathbf{w}) = \mathbf{w}^T(\mathbf{u} \times \mathbf{v})$ – this is called the “scalar triple product.” Therefore,

$$\begin{aligned} -\mathbf{V}_i^T \mathbf{M}(\mathbf{u}_i) \mathbf{h} &= \begin{bmatrix} -\mathbf{v}_i^T(\mathbf{u}_i \times \mathbf{h}) \\ -\mathbf{w}_i^T(\mathbf{u}_i \times \mathbf{h}) \end{bmatrix} = \begin{bmatrix} -\mathbf{h}^T(\mathbf{v}_i \times \mathbf{u}_i) \\ -\mathbf{h}^T(\mathbf{w}_i \times \mathbf{u}_i) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{h}^T \mathbf{w}_i \\ -\mathbf{h}^T \mathbf{v}_i \end{bmatrix} = \mathbf{X}_i \mathbf{h}. \end{aligned}$$

and

$$\rho_2(\hat{\mathbf{A}}) \approx \sum \|\tilde{\mathbf{e}}_i - \mathbf{X}_i \mathbf{h}\|^2 = \sum \|\mathbf{y}_i - \mathbf{X}_i \mathbf{h}\|^2 \quad (53.28)$$

where $\mathbf{y}_i = \tilde{\mathbf{e}}_i$, a 2×1 vector.

\mathbf{y}_i is distributed bivariate normal $N(\mathbf{X}_i \mathbf{0}, \kappa^{-1} \mathbf{I}_2)$. In other words, \mathbf{y}_i satisfies the assumptions of the regression model

$$\mathbf{y}_i = \mathbf{X}_i \boldsymbol{\beta} + \boldsymbol{\epsilon}_i$$

$$\boldsymbol{\epsilon}_i \text{ i.i.d. } N(\mathbf{0}, \kappa^{-1} \mathbf{I}_2)$$

where the “true” value of $\boldsymbol{\beta}$ is $\mathbf{0}$. The right hand side of (53.28) is the sum squares to be minimized by this regression, where $\hat{\mathbf{A}} = \mathbf{A} \Phi(\mathbf{h})$ and the minimum occurs at $\hat{\boldsymbol{\beta}} = \mathbf{h}$.

Let the $2n \times 3$ matrix \mathbf{X} be obtained by stacking $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n$ on top of each other. Then, using (53.27)

$$\mathbf{X}^T \mathbf{X} = \sum (\mathbf{w}_i \mathbf{w}_i^T + \mathbf{v}_i \mathbf{v}_i^T) = n(\mathbf{I}_3 - \boldsymbol{\Sigma})$$

where $\boldsymbol{\Sigma} = n^{-1} \sum \mathbf{u}_i \mathbf{u}_i^T$. Standard regression theory then indicates that the (asymptotic) distribution of \mathbf{h} is thus $N(\boldsymbol{\beta}, \sigma^2(\mathbf{X}^T \mathbf{X})^{-1})$ or, in this case, $N(\mathbf{0}, (n\kappa(\mathbf{I}_3 - \boldsymbol{\Sigma}))^{-1})$.

In summary, for example, to test $H_0 : \mathbf{A} = \mathbf{A}_0$, calculate the $\hat{\mathbf{A}}$ that minimizes (53.6). Find \mathbf{h} so that $\mathbf{A}_0^T \hat{\mathbf{A}} = \Phi(\mathbf{h})$ and compare $n\hat{\kappa} \mathbf{h}^T (\mathbf{I}_3 - \boldsymbol{\Sigma}) \mathbf{h}$ to a $3F_{3, 2n-3}$ distribution, that is, $n\hat{\kappa} \mathbf{h}^T (\mathbf{I}_3 - \boldsymbol{\Sigma}) \mathbf{h}/3$ is compared to an F-distribution with 3 and $2n - 3$ degrees of freedom. Alternatively, a confidence region for \mathbf{A} is $\{\hat{\mathbf{A}} \Phi(\mathbf{h}) \mid n\hat{\kappa} \mathbf{h}^T (\mathbf{I}_3 - \boldsymbol{\Sigma}) \mathbf{h} < 3F_{3, 2n-3, \alpha}\}$.

Note that in our approximating regression, from (53.28), $\rho_2(\hat{\mathbf{A}})$ plays the role of the error sum squares. So we can use the estimate $\hat{\kappa}^{-1} = \rho_2(\hat{\mathbf{A}}) / (2n - 3)$.

Finally we note that if $\boldsymbol{\Sigma} = \lambda_1 \mathbf{e}_1 \mathbf{e}_1^T + \lambda_2 \mathbf{e}_2 \mathbf{e}_2^T + \lambda_3 \mathbf{e}_3 \mathbf{e}_3^T$ is an eigendecomposition of $\boldsymbol{\Sigma}$, then $\lambda_1 + \lambda_2 + \lambda_3 = \text{Tr}(\boldsymbol{\Sigma}) = \text{Tr}(n^{-1} \sum \mathbf{u}_i \mathbf{u}_i^T) = n^{-1} \sum \text{Tr}(\mathbf{u}_i \mathbf{u}_i^T) = n^{-1} \sum \text{Tr}(\mathbf{u}_i^T \mathbf{u}_i) = 1$, using (53.25), and hence the eigenvalues of $\mathbf{I}_3 - \boldsymbol{\Sigma}$ are $\lambda_2 + \lambda_3, \lambda_1 + \lambda_3, \lambda_1 + \lambda_2$. Thus,

$$\begin{aligned} (\mathbf{I}_3 - \boldsymbol{\Sigma})^{-1} &= (\lambda_2 + \lambda_3)^{-1} \mathbf{e}_1 \mathbf{e}_1^T + (\lambda_1 + \lambda_3)^{-1} \mathbf{e}_2 \mathbf{e}_2^T \\ &\quad + (\lambda_1 + \lambda_2)^{-1} \mathbf{e}_3 \mathbf{e}_3^T. \end{aligned}$$

Theorem 31.4.4 and the results of Sect. 31.4.9 of [1] are expressed using this identity.

Ordering the eigenvalues and eigenvectors so that $\lambda_1 \geq \lambda_2 \geq \lambda_3$, \mathbf{e}_1 will be the center of the \mathbf{u}_i , and \mathbf{e}_3 will be perpendicular the great circle that best fits the \mathbf{u}_i . Then \mathbf{h} will have the greatest variability in the direction \mathbf{e}_1 and the least in the direction \mathbf{e}_3 . In the plate tectonic context, using heuristic reasoning, Stock and Molnar (see [12, 13]) predicted this behavior.

53.6 The Reconstruction of Tectonic Plate Motions from Magnetic Anomaly Lineations

Mathematically, the approach used in the derivations in Sects. 53.2.1, 53.5.1, and 53.5.2 was to use (first order) Taylor series approximations together with appropriately chosen tangent space approximations. The result was an approximating regression problem which is then analyzed using elementary least squares (regression) results. However, in Sect. 53.5.2, each \mathbf{v}_i has its own basepoint $\mathbf{A} \mathbf{u}_i$ for a tangent space approximation, and the resulting regression problem took place in multiple tangent spaces. Some matrix manipulation was necessary to “unify the basepoints” before applying standard regression results. This matrix manipulation, although tedious, is routine and masks the actually simple nature of the arguments in Sect. 53.5.2.

The spherical regression model assumes an idealized form of the data that, in fact, does not exist in plate tectonics. We discuss in this section the reconstruction of tectonic plate motions using experimentally measured locations on “marine magnetic anomaly lineations.” As we shall see the statistical justification for the proposed methodology closely follows the general outline of the previous paragraph. This discussion is based upon [14].

Marine magnetic anomaly lineation data is used to estimate the relative motion of two plates that are diverging at a mid-oceanic ridge. Molten lava is extruded at or near the ridge axis and hardens essentially immediately after extrusion. The new crustal material moves away from the ridge, carried by the plates, as if on diverging conveyer belts. If we could identify on each plate the material that was extruded at the same time, we would have two isochrons, one on each side of the ridge. The two paired isochrons would be congruent, each having the shape of the ridge at the common time of their extrusion. Since the two isochrons are congruent, there is a rotation \mathbf{A} so that if \mathbf{u} is a point on the U-plate isochron, then $\mathbf{A} \mathbf{u}$ would be (in the absence of

error) the homologous point on the V-plate isochron. Thus, if we could identify homologous pairs $(\mathbf{u}_i, \mathbf{v}_i)$, they could be used in a spherical regression to estimate \mathbf{A} and statistically assess the errors in the estimate. $\hat{\mathbf{A}}^T$ is said to reconstruct the past (at the common time of extrusion) position of the V-plate to the U-plate in a coordinate system fixed in the U-plate.

A very helpful figure due to Muller et al. [15] is available at https://commons.wikimedia.org/wiki/File:Age_of_oceanic_lithosphere.jpg.

It shows the mid-Atlantic ridge (solid black line) between South America and Africa. Regions of crustal material of the same color were extruded from the ridge in the same time interval and have the shape of the ridge at the common time of their extrusion. They occur in matching congruent pairs, one on each side of the ridge.

Careful examination of the shape of the mid-Atlantic ridge in Muller et al. [15] shows that it has a rather piecewise linear shape. The dominant direction of the segments of this piecewise linear shape is roughly parallel to the ridge. However, there are cross segments that are due to “fracture zones.” On the sea floor bottom, fracture zones look like valleys and can be detected by bathymetric measurement.

Figure 53.1 is a stylized diagram of a mid-oceanic ridge (solid black line) with two pairs of matching isochrons, dated 10 and 20 million years before present (mybp), one isochron of each pair on each side of the ridge. Two isochrons of the same age are congruent. However, the shape of the ridge does change with time, so isochrons of different ages are not necessarily congruent. A hypothetical homologous pair $(\mathbf{u}_1, \mathbf{v}_1)$ is also shown.

The cooling solid rock acquires a direction of magnetization that is determined by the location of the north magnetic pole at the time of cooling below the “blocking tempera-

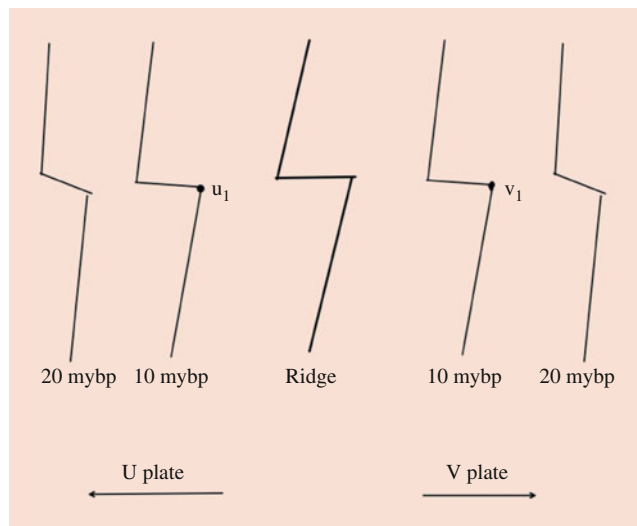


Fig. 53.1 Stylized diagram of a mid-oceanic ridge with two pairs of matching isochrons

ture of magnetization.” At specific times in the past, the “magnetic anomalies,” the north magnetic pole shifted to the geographic south. For example, the A6 anomaly occurred approximately 20 million years ago. Research vessels can measure the sea floor’s magnetic field as they cross the ocean. After subtracting the effects of the dominant magnetization (of the current magnetic field), a skilled geophysicist can determine when the vessel crossed the isochron of a specific magnetic anomaly. Thus, the data consists of identified locations of crossings of the magnetic anomaly lineation.

A research vessel is unlikely to collect crossing locations of opposing isochrons in homologous pairs. Thus, the spherical regression model does not apply.

53.6.1 The Hellinger Criterion to Fit Estimate a Reconstruction

Hellinger [16] proposed an approach for estimating \mathbf{A} . In the Hellinger framework, the isochrons are modeled as piecewise great circle segments. Let $\boldsymbol{\eta}_1, \boldsymbol{\eta}_2, \dots, \boldsymbol{\eta}_s$ denote the normal vectors to the great circle segments on the U-side isochron. Then $\mathbf{A}\boldsymbol{\eta}_1, \mathbf{A}\boldsymbol{\eta}_2, \dots, \mathbf{A}\boldsymbol{\eta}_s$ are the normal vectors to the great circle segments on the V-side isochron. Let $\mathbf{u}_{ij}, i = 1, \dots, s, j = 1, \dots, m_i$ denote the estimated crossings of the U-side isochron; here i indexes the isochron segment. Let $\mathbf{v}_{ik}, i = 1, \dots, s, k = 1, \dots, n_i$ denote the estimated crossings on the V-side isochron. The crossings on the U-side are not paired with those on the V-side so m_i and n_i are generally different. See Fig. 53.2.

The isochron crossings can be identified with differing levels of precision. Each magnetic anomaly has a characteristic squiggle that should be reflected in the magnetic record of the ship’s trajectory. The Hellinger methodology allows the scientist to assign uncertainty constants κ_{ij} and $\tilde{\kappa}_{ik}$ to each estimated crossing, based upon, for example, the difficulty of identifying the different characteristic squiggles. In essence, subject to an overall unknown “fudge factor” κ , $(\kappa \kappa_{ij})^{-1/2}$ represents the standard deviation of the error of \mathbf{u}_{ij} in the direction $\boldsymbol{\eta}_i$. $\tilde{\kappa}_{ik}$ has a similar interpretation.

Hellinger proposed to simultaneously estimate \mathbf{A} and the $\boldsymbol{\eta}_i$ by minimizing the objective function

$$r(\mathbf{A}, \boldsymbol{\eta}_1, \boldsymbol{\eta}_2, \dots, \boldsymbol{\eta}_s) = \sum_{i,j} \kappa_{ij} (\mathbf{u}_{ij}^T \boldsymbol{\eta}_i)^2 + \sum_{i,k} \tilde{\kappa}_{ik} (\mathbf{v}_{ik}^T \mathbf{A} \boldsymbol{\eta}_i)^2. \quad (53.29)$$

Notice that (53.29) is a weighted sum of the square distances from the identified crossings to the corresponding isochron segments.

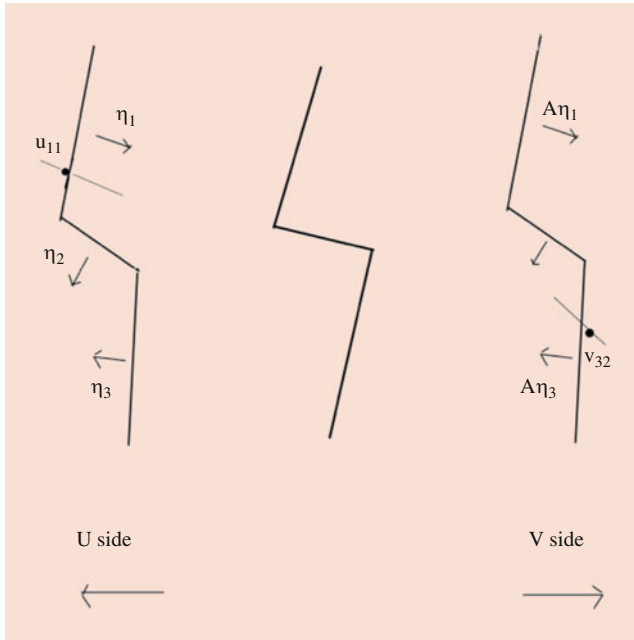


Fig. 53.2 U-side and V-side isochrons with present day ridge (darkest). Section normal vectors $\eta_1, \eta_2, \eta_3, A\eta_1, A\eta_2, A\eta_3$. Ship trajectory (lightest line), crossings u_{11} (U-side segment 1), v_{32} (V-side segment 3)

53.6.2 Statistical Analysis of the Hellinger Estimate

We will use Taylor linearization together with tangent space approximations to $SO(3)$ and to S^2 at each u_{ij}, v_{ik} to build an approximating regression for which (53.29) is asymptotically equivalent to the regression's sum of squares.

The errors in this type of data are quite small, around 10 km, so large κ approximations are fully appropriate. Thus, we use separate local “flat Earth” approximations at each data point u_{ij}, v_{ik} . The total reach of the isochrons is quite extensive, so using a global flat Earth mapping (such as latitude and longitude) would introduce substantial distortion.

We assume that u_{ij}, v_{ik} are measurements of “true” locations $\alpha_{ij}, \tilde{\alpha}_{ik}$ with $\alpha_{ij}^T \eta_i = 0, \tilde{\alpha}_{ik}^T A\eta_i = 0$. Choose unit vectors $\beta_{ij}, \tilde{\beta}_{ik}$ so that $\alpha_{ij}, \beta_{ij}, \eta_i$ and $\tilde{\alpha}_{ik}, \tilde{\beta}_{ik}, A\eta_i$ form right-hand-rule-oriented orthonormal coordinate systems of R^3 . Let $\lambda_{ij}, \tilde{\lambda}_{ik}$ be fixed “tangential” error constants.

In the data space write the tangential approximations

$$\begin{aligned} u_{ij} &= (u_{ij}^T \alpha_{ij}) \alpha_{ij} + \epsilon_{ij} \approx \alpha_{ij} + \epsilon_{ij}, \\ v_{ik} &= (v_{ik}^T \tilde{\alpha}_{ik}) \tilde{\alpha}_{ik} + \tilde{\epsilon}_{ik} \approx \tilde{\alpha}_{ik} + \tilde{\epsilon}_{ik} \end{aligned} \quad (53.30)$$

where $\epsilon_{ij}^T \alpha_{ij} = 0, \tilde{\epsilon}_{ik}^T \tilde{\alpha}_{ik} = 0$. We assume that as $\kappa \rightarrow \infty$, the vectors $\epsilon_{ij} \in \alpha_{ij}^\perp$ have bivariate normal distributions with mean 0 and variance covariance matrices $(\kappa \kappa_{ij})^{-1} \eta_i \eta_i^T + (\kappa \lambda_{ij})^{-1} \beta_{ij} \beta_{ij}^T$. In essence, α_{ij}^\perp is spanned by η_i and β_{ij} and we are assuming that the component of ϵ_{ij}

in the direction η_i has variance $(\kappa \kappa_{ij})^{-1}$ and its component in the direction β_{ij} has variance $(\kappa \lambda_{ij})^{-1}$. Similar assumptions are made about the distribution of the $\tilde{\epsilon}_{ik} \in \tilde{\alpha}_{ik}^\perp$.

The parameters in this problem are $A \in SO(3)$ and the $\eta_i \in S^2$. Tangentially approximate their estimates by

$$\hat{\eta}_i \approx \eta_i + \xi_i, \quad \hat{A} = A\Phi(\mathbf{h}) \approx A(\mathbf{I} + \mathbf{H}) \quad (53.31)$$

where $\xi_i \in \eta_i^\perp$, \mathbf{H} is 3×3 skew symmetric whose components make up the 3-vector \mathbf{h} using the conventions of the previous section or equation (53.14). We seek the distribution of \mathbf{h} . Substituting (53.30) and (53.31), using $\eta_i^T \alpha_{ij} = 0, \eta_i^T A^T \tilde{\alpha}_{ik} = 0$, and ignoring lower order terms

$$\begin{aligned} r(\hat{A}, \hat{\eta}_1, \hat{\eta}_2, \dots, \hat{\eta}_s) &\approx \sum_{i,j} \kappa_{ij} (\alpha_{ij}^T \xi_i + \epsilon_{ij}^T \eta_i)^2 \\ &+ \sum_{i,k} \tilde{\kappa}_{ik} (\tilde{\alpha}_{ik}^T A \xi_i + \tilde{\alpha}_{ik}^T A \mathbf{H} \eta_i + \tilde{\epsilon}_{ik}^T A \eta_i)^2 \end{aligned} \quad (53.32)$$

Furthermore,

$$\begin{aligned} \tilde{\alpha}_{ik}^T A \mathbf{H} \eta_i &= (A^T \tilde{\alpha}_{ik})^T \mathbf{H} \eta_i = \mathbf{h}^T (\eta_i \times (A^T \tilde{\alpha}_{ik})) \\ &= \mathbf{h}^T (A^T \tilde{\beta}_{ik}) = (A^T \tilde{\beta}_{ik})^T \mathbf{h} \end{aligned}$$

By assumption $y_{ij} = \epsilon_{ij}^T \eta_i$ and $\tilde{y}_{ik} = \tilde{\epsilon}_{ik}^T A \eta_i$ are distributed (univariate) $N(0, (\kappa \kappa_{ij})^{-1})$ and $N(0, (\kappa \tilde{\kappa}_{ik})^{-1})$, respectively.

Thus (53.32) is the error sum squares of the weighted regression model

$$y_{ij} = -\alpha_{ij}^T \xi_i + \epsilon_{ij}$$

$$\tilde{y}_{ik} = -\tilde{\alpha}_{ik}^T A \xi_i - (A^T \tilde{\beta}_{ik})^T \mathbf{h} + \tilde{\epsilon}_{ik}.$$

Referring to the standard form of the regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

design matrix \mathbf{X} has $\sum(m_i + n_i)$ rows, one for each crossing point. The vector $\boldsymbol{\beta}$ has $3 + 2s$ entries, 3 for \mathbf{h} and 2 for each ξ_i . It is actually an error vector so its “true” value is $\mathbf{0}$. The assigned uncertainties $\kappa_{ij}, \tilde{\kappa}_{ik}$ are the diagonal entries of the weight matrix \mathbf{W} .

In fact there is no \mathbf{y} vector and no fitted values of $\boldsymbol{\beta}$. But we can estimate its covariance matrix $\kappa^{-1}(\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1}$ by estimating κ^{-1} from $r(\hat{A}, \hat{\eta}_1, \hat{\eta}_2, \dots, \hat{\eta}_s)$ that plays the role of the regression residual sum squares, and $\mathbf{X}^T \mathbf{W} \mathbf{X}$ from the data. That, together with the estimates $\hat{A}, \hat{\eta}_i$ obtained by minimizing (53.29), is sufficient for statistical procedures.

At this point it is easier to write a program to implement the procedure than to explain in detail what is actually done. [17] describes a package of programs for plate reconstructions.

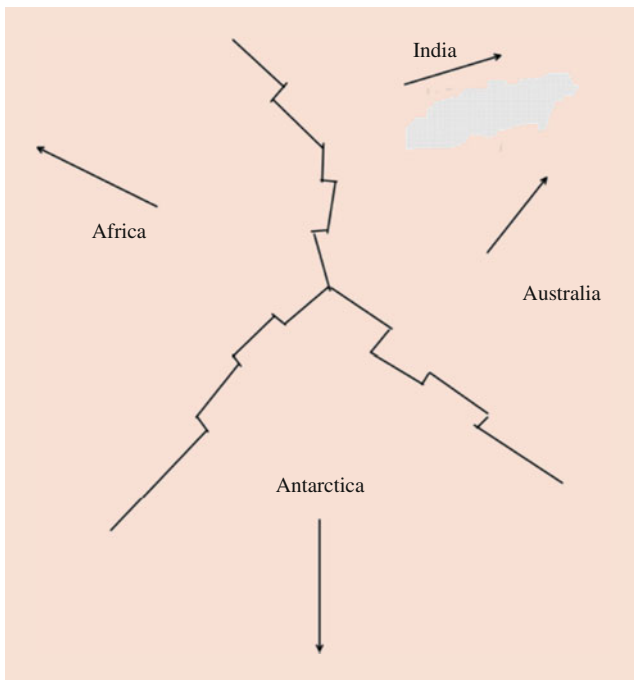


Fig. 53.3 Stylized diagram of Indian Ocean triple junction, after Royer & Chang [18]. Stippled region is area of possible (slow) convergence between Australia and India

53.6.3 Additional Work Using Hellinger Estimates

Examining Muller et al. [15], one can see a triple junction of three diverging plate boundaries in the Indian Ocean. A stylized picture of the geometry of the triple junction (between Africa, Antarctica, and Australia) is shown in Fig. 53.3. The regression model of the previous section can be extended to analyze triple junctions.

In addition, if two plates do not share a diverging boundary, one can calculate their relative motion by combining rotations to a third plate with divergent boundaries to both plates: If \mathbf{A} takes the U-plate to the V-plate and \mathbf{B} takes the U-plate to the W-plate, then $\mathbf{C} = \mathbf{B}\mathbf{A}^T$ takes the V-plate to the W-plate and one can estimate, again through Taylor linearization, the errors in $\hat{\mathbf{C}} = \hat{\mathbf{B}}\hat{\mathbf{A}}^T$ from those of its constituent estimated rotations.

In particular, let $\hat{\mathbf{A}} = \mathbf{A}\Phi(\mathbf{h}_A)$, $\hat{\mathbf{B}} = \mathbf{B}\Phi(\mathbf{h}_B)$ and write $\hat{\mathbf{C}} = \mathbf{C}\Phi(\mathbf{h}_C)$. Then

$$\begin{aligned} \hat{\mathbf{C}} &= \hat{\mathbf{B}}\hat{\mathbf{A}}^T = \mathbf{B}\Phi(\mathbf{h}_B)(\mathbf{A}\Phi(\mathbf{h}_A))^T \\ &\approx \mathbf{B}(\mathbf{I}_3 + \mathbf{M}(\mathbf{h}_B))(\mathbf{I}_3 + \mathbf{M}(-\mathbf{h}_A))\mathbf{A}^T \\ &= \mathbf{B}(\mathbf{I}_3 + \mathbf{M}(\mathbf{h}_B))\mathbf{A}^T(\mathbf{I}_3 + \mathbf{M}(-\mathbf{A}\mathbf{h}_A)) \\ &= \mathbf{B}\mathbf{A}^T(\mathbf{I}_3 + \mathbf{M}(\mathbf{A}\mathbf{h}_B))(\mathbf{I}_3 + \mathbf{M}(-\mathbf{A}\mathbf{h}_A)) \\ &\approx \mathbf{C}(\mathbf{I}_3 + \mathbf{M}(\mathbf{A}(\mathbf{h}_B - \mathbf{h}_A))) \end{aligned} \quad (53.33)$$

Therefore,

$$\mathbf{h}_C \approx \mathbf{A}(\mathbf{h}_B - \mathbf{h}_A). \quad (53.34)$$

Since $\mathbf{h}_A, \mathbf{h}_B$ are both trivariate normal with mean $\mathbf{0}$ and are independent (since it is physically impossible for the same crossing to be on both boundaries), it suffices to estimate

$$\widehat{\text{Cov}}(\mathbf{h}_C) = \hat{\mathbf{A}}(\widehat{\text{Cov}}(\mathbf{h}_A) + \widehat{\text{Cov}}(\mathbf{h}_B))\hat{\mathbf{A}}^T \quad (53.35)$$

Statistical error analysis is needed to discern if small misfits in plate reconstructions are genuine or due to “statistical error.” Using these techniques, [18] determined that the Indo-Australian plate is actually two component plates with a relatively small converging internal movement. It is relevant to note that an area of increased seismic activity and seafloor undulations indicate where the boundary between India and Australia should be if the two component plates do not move rigidly together.

The rotation of India to Australia was estimated by combining the estimated rotations $\hat{\mathbf{A}}$ of Africa (U-plate) to India (V-plate) and $\hat{\mathbf{B}}$ of Africa to Australia (W-plate). The latter was constrained by the Indian Ocean triple junction. See again Fig. 53.3.

Now letting $\hat{\mathbf{C}} = \hat{\mathbf{B}}\hat{\mathbf{A}}^T$, one solves $\hat{\mathbf{C}} = \Phi(\mathbf{h}_C)$ for \mathbf{h}_C . Using (53.35) to estimate $\text{Cov}(\mathbf{h}_C)$. [18] found that $\mathbf{h}_C^T[\widehat{\text{Cov}}(\mathbf{h}_C)]^{-1}\mathbf{h}_C$ is significant compared to a χ_3^2 distribution and concluded that \mathbf{C} is not the identity, or equivalently, $\mathbf{A} \neq \mathbf{B}$.

Since then the Indo-Australian plate has been modeled as a composite plate with 4 component plates (Indian, Capricorn, Australian, and Macquarie), see for example DeMets, Gordon, and Royer [19]. In essence they propose modeling nonrigidities in a plate by using a collection of small component plates that move slowly relatively to each other and slower than the overall motion of the ambient larger plate. The same approach, and the software in [17], was used.

Once one is able to produce good approximating regressions, one can create spherical regression analogs of existing linear regression tools. For example, a straight forward use of weighted linear regression requires that all uncertainty constants $\kappa_{ij}, \tilde{\kappa}_{ik}$ be assigned with a common fudge factor κ . However, different fudge factors might be preferred, especially when working with multiple plate boundaries. This is essentially a spherical regression version of the classical Behrens-Fisher problem. Kirkwood [20] applied the solution of Johansen [21] for the analogous problem in the linear regression setting to solve the problem of multiple κ 's.

The Hellinger model assumes that the lineations are composed of great circle segments. As data becomes more numerous, this approximation can be called into question. Some preliminary work by H. Hendriks, R. Yang, and Chang assumed that the deviations of these segments from a great

circle can be modeled as a Brownian bridge. This work is incomplete, but the author believes that curved segment model is a problem that should soon be addressed.

53.7 Summary

In geometric statistical problems, the parameter is often a manifold. When employing Taylor series methods to study the asymptotic behavior of parameter estimates, it is necessary to map (reparameterize) the manifold into Euclidean space. Rather than reparameterizing the manifold globally, it is often better to use a local parameterization from a tangent space approximation.

When the data is also non-Euclidean, a data transformation of the data space to globally turn it into Euclidean space is usually inappropriate. Rather, when the errors in the data are small relative to the curvature of the data space, “local flat Earth” approximations are appropriate and, in this case, tangent space approximations to the data space are called for. With “local flat Earth” approximation, an approximating linear regression can often be found and the problem can be studied using standard regression methodology.

This chapter has described the underlying mathematical constructs and applied this process to four problems of increasing complexity.

Acknowledgments The author wishes to acknowledge the encouragement of Prof. Sang-mook Lee, a geophysics professor at Seoul National University. Prof. Lee admonished the author that it is his “scientific duty” to explain, in as elementary manner as possible, the mathematics underlying his work. His dedication to science and his cheerfulness inspire all who have met him. This chapter has been written to honor him.



Prof. Sang-Mook Lee and author (standing)
June 20, 2013
Manassas, VA

Appendix

In this section, we outline the derivation of the formula for the errors in the estimate $\hat{\mathbf{C}} = \hat{\mathbf{B}}\hat{\mathbf{A}}^T$ of a combined rotation $\mathbf{C} = \mathbf{B}\mathbf{A}^T$ using the commonly used parameterization (axis latitude, axis longitude, angle of rotation), or (λ, θ, ρ) , of a rotation matrix in $\text{SO}(3)$. In other words, we derive the analogue to (53.35) using (λ, θ, ρ) and restrict ourselves to the case $k = 3$.

This parameterization is geometrically unnatural. We will see that, as mentioned in Sect. 53.4.2, ignoring the geometry of $\text{SO}(3)$ leads to a far messier version of (53.35).

Quaternion Representation of $\text{SO}(3)$

To work in the parameters (λ, θ, ρ) of a rotation, it is easier to use the quaternion representation of a rotation rather than its matrix representation.

The quaternions are a four dimensional extension of the real numbers in the same sense that the complex numbers are a two-dimensional extension of the reals. In particular, the quaternions are $\mathbb{Q} = \{\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \mid q_0, q_1, q_2, q_3 \in \mathbb{R}\}$. \mathbf{i}, \mathbf{j} , and \mathbf{k} are special quaternions that play a similar role in \mathbb{Q} to the role of i in the complex numbers.

Multiplication in \mathbb{Q} is defined by the rules

$$\begin{aligned} i^2 = j^2 = k^2 = -1, \quad ij = -ji = k, \quad jk = -kj = i, \\ ki = -ik = j. \end{aligned} \quad (53.36)$$

\mathbb{Q} satisfies all the usual rules of algebra except that, as apparent in (53.36), multiplication is not commutative.

If $\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \in \mathbb{Q}$, its “conjugate” is $\bar{\mathbf{q}} = q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k}$. Thus, the unit sphere $S^3 \subset \mathbb{R}^4$ can be represented as the set of “unit length quaternions,” that is, those quaternions $\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$ that satisfy

$$\mathbf{q}\bar{\mathbf{q}} = \bar{\mathbf{q}}\mathbf{q} = q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1.$$

Indeed it is easily checked that the Euclidean \mathbb{R}^4 dot product between quaternions \mathbf{q}_1 and \mathbf{q}_2 is $(\bar{\mathbf{q}}_1\mathbf{q}_2 + \bar{\mathbf{q}}_2\mathbf{q}_1) / 2$. Note also that $\bar{\mathbf{q}}_1\bar{\mathbf{q}}_2 = \bar{\mathbf{q}}_2\bar{\mathbf{q}}_1$.

So if a rotation \mathbf{A} has axis latitude, axis longitude, and angle of rotation (λ, θ, ρ) , its quaternion representation is

$$\mathbf{q} = \cos \frac{\rho}{2} + \sin \frac{\rho}{2} (\sin \lambda \cos \theta \mathbf{i} + \sin \lambda \sin \theta \mathbf{j} + \cos \lambda \mathbf{k}). \quad (53.37)$$

The “pure quaternions” can be identified as $\mathbb{R}^3 \approx \{\tilde{\mathbf{x}} = x_1\mathbf{i} + x_2\mathbf{j} + x_3\mathbf{k}\}$. In this way, we see that $\sin(\lambda)$

$\cos(\theta)\mathbf{i} + \sin(\lambda)\sin(\theta)\mathbf{j} + \cos(\lambda)\mathbf{k}$ is the pure quaternion representation of the axis of \mathbf{A} .

In general, if $\mathbf{x} \in \mathbf{R}^3$ is represented as a 3×1 vector, then we will denote its representation as a pure quaternion by $\tilde{\mathbf{x}}$. It can easily be verified that the pure quaternion representation of \mathbf{Ax} is $q\tilde{\mathbf{x}}\bar{q}$.

It is clear that q is a unit length quaternion and that q and $-q$ represent the same rotation. Mathematically we say that $\text{SO}(3)$ is “doubly covered” by S^3 .

This double covering of $\text{SO}(3)$ by S^3 is a disguised form of the long known “Cayley-Klein” parameters. For $q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$, let

$$\mathbf{H}(q) = \begin{bmatrix} q_0 - q_3i & -q_2 - q_1i \\ q_2 - q_1i & q_0 + q_3i \end{bmatrix}. \quad (53.38)$$

It is easily checked that $\det(\mathbf{H}(q)) = 1$ and that $\mathbf{H}(q)\overline{\mathbf{H}(q)}^T = \mathbf{I}_2$. The collection of 2×2 complex matrices \mathbf{A} that satisfy $\det(\mathbf{A}) = 1$ and $\mathbf{A}\bar{\mathbf{A}}^T = \mathbf{I}_2$ is called the “special unitary” group, denoted by $\text{SU}(2)$ and (53.38) identifies S^3 with $\text{SU}(2)$. Thus, $\mathbf{H}(q) \rightarrow \mathbf{A}$ defines a double covering of $\text{SO}(3)$ by $\text{SU}(2)$. For a more classical description of the Cayley-Klein parameters, see, for example, Goldstein [22].

Combining Two Rotations Using Quaternionic Representation

The tangent vectors to S^3 at 1, considered as a unit length quaternion and hence an element of S^3 , are the pure quaternions. Hence, the tangent vectors at $q \in S^3$ have the form $q\tilde{\mathbf{x}}$ where $\tilde{\mathbf{x}}$ is pure quaternion. Hence, we write

$$q_A \approx q + q_A\tilde{\mathbf{x}}_A \quad (53.39)$$

where $\tilde{\mathbf{x}}_A$ is pure quaternion.

If $\mathbf{C} = \mathbf{BA}^T$, then $q_C = q_B\bar{q}_A$. Thus,

$$\begin{aligned} q_C + q_C\tilde{\mathbf{x}}_C &\approx q_C\tilde{\mathbf{x}}_C \approx q_B\bar{q}_A\tilde{\mathbf{x}}_C \approx (q_B + q_B\tilde{\mathbf{x}}_B)(\bar{q}_A - \tilde{\mathbf{x}}_A\bar{q}_A) \\ &\approx q_B\bar{q}_A + q_B\tilde{\mathbf{x}}_B\bar{q}_A - q_B\tilde{\mathbf{x}}_A\bar{q}_A \end{aligned}$$

$$\tilde{\mathbf{x}}_C \approx \bar{q}_C(q_B\tilde{\mathbf{x}}_B\bar{q}_A - q_B\tilde{\mathbf{x}}_A\bar{q}_A) = q_A(\tilde{\mathbf{x}}_B - \tilde{\mathbf{x}}_A)\bar{q}_A. \quad (53.40)$$

(53.40) is the quaternionic version of (53.34). Indeed, by carefully following all the identifications in section “Quaternion Representation of $\text{SO}(3)$ ”, it can be shown that if we also write $\hat{\mathbf{A}} = \mathbf{A}\Phi(\mathbf{h}_A)$, then the 3-vector representation of $\tilde{\mathbf{x}}_A$ is $\mathbf{x}_A = \mathbf{h}_A/2$, and hence (53.40) would follow from (53.34).

Combining Two Rotations Using Axis Latitude, Longitude, and Angle of Rotation

We now compute the change of variables from $[\hat{\lambda} - \lambda \hat{\theta} - \theta \hat{\rho} - \rho]^T$ to \mathbf{x} . We will apply the Taylor approximations $\sin\left(\frac{\rho}{2}\right) \approx \sin\left(\frac{\rho}{2}\right) + \cos\left(\frac{\rho}{2}\right)(\hat{\rho} - \rho)/2$ and $\cos\left(\frac{\rho}{2}\right) \approx \cos\left(\frac{\rho}{2}\right) - \sin\left(\frac{\rho}{2}\right)(\hat{\rho} - \rho)/2$ and similar approximations for λ, θ in (53.37) to arrive at

$$\begin{aligned} \hat{q} &\approx q - \sin\frac{\rho}{2}\frac{(\hat{\rho}-\rho)}{2} \\ &\quad + \cos\frac{\rho}{2}\frac{(\hat{\rho}-\rho)}{2}(\sin\lambda\cos\theta\mathbf{i} + \sin\lambda\sin\theta\mathbf{j} + \cos\lambda\mathbf{k}) \\ &\quad + \sin\frac{\rho}{2}\cos\lambda(\hat{\lambda} - \lambda)(\cos\theta\mathbf{i} + \sin\theta\mathbf{j}) \\ &\quad + \sin\frac{\rho}{2}\sin\lambda(-\sin\theta\mathbf{i} + \cos\theta\mathbf{j})(\hat{\theta} - \theta) \\ &\quad - \sin\frac{\rho}{2}\sin\lambda(\hat{\lambda} - \lambda)\mathbf{k}. \end{aligned}$$

So

$$\hat{q} - q \approx \begin{bmatrix} \sin\frac{\rho}{2}(\cos\lambda\cos\theta\mathbf{i} + \cos\lambda\sin\theta\mathbf{j} - \sin\lambda\mathbf{k}) \\ \sin\frac{\rho}{2}(-\sin\lambda\sin\theta\mathbf{i} + \sin\lambda\cos\theta\mathbf{j}) \\ -\frac{1}{2}\sin\frac{\rho}{2} + \frac{1}{2}\cos\frac{\rho}{2}(\sin\lambda\cos\theta\mathbf{i} + \sin\lambda\sin\theta\mathbf{j} + \cos\lambda\mathbf{k}) \end{bmatrix}^T \begin{bmatrix} \hat{\lambda} - \lambda \\ \hat{\theta} - \theta \\ \hat{\rho} - \rho \end{bmatrix}$$

and, after substituting (53.37) and substantial simplification,

$$\tilde{\mathbf{x}} = \bar{q}(\hat{q} - q) \approx \begin{bmatrix} F_{11}\mathbf{i} + F_{21}\mathbf{j} + F_{31}\mathbf{k} \\ F_{12}\mathbf{i} + F_{22}\mathbf{j} + F_{32}\mathbf{k} \\ F_{13}\mathbf{i} + F_{23}\mathbf{j} + F_{33}\mathbf{k} \end{bmatrix}^T \begin{bmatrix} \hat{\lambda} - \lambda \\ \hat{\theta} - \theta \\ \hat{\rho} - \rho \end{bmatrix} \quad (53.41)$$

where

$$F_{11} = \sin\frac{\rho}{2}\left(\cos\frac{\rho}{2}\cos\lambda\cos\theta + \sin\frac{\rho}{2}\sin\theta\right)$$

$$F_{21} = \sin\frac{\rho}{2}\left(\cos\frac{\rho}{2}\cos\lambda\sin\theta - \sin\frac{\rho}{2}\cos\theta\right)$$

$$F_{31} = -\sin\frac{\rho}{2}\cos\frac{\rho}{2}\sin\lambda$$

$$F_{12} = -\sin\frac{\rho}{2}\sin\lambda\left(\cos\frac{\rho}{2}\sin\theta - \sin\frac{\rho}{2}\cos\lambda\cos\theta\right)$$

$$F_{22} = \sin\frac{\rho}{2}\sin\lambda\cos\theta\left(\cos\frac{\rho}{2} + \sin\frac{\rho}{2}\cos\lambda\right)$$

$$\begin{aligned}
 F_{32} &= -\left(\sin \frac{\rho}{2} \sin \lambda\right)^2 \\
 F_{13} &= \frac{1}{2} \sin \lambda \cos \theta \\
 F_{23} &= \frac{1}{2} \sin \lambda \sin \theta \\
 F_{33} &= \frac{1}{2} \cos \lambda.
 \end{aligned} \tag{53.42}$$

Rewriting (53.41) in real vector-matrix form

$$\mathbf{x} = \mathbf{F}(\lambda, \theta, \rho) \begin{bmatrix} \hat{\lambda} - \lambda \\ \hat{\theta} - \theta \\ \hat{\rho} - \rho \end{bmatrix}$$

where $\mathbf{F}(\lambda, \theta, \rho)$ is the 3×3 matrix with real entries F_{ij} .

Recall that the pure quaternion representation of \mathbf{Ax} is $q\tilde{x}\bar{q}$. Thus, using (53.40)

$$\mathbf{x}_C \approx \mathbf{A}(\mathbf{x}_B - \mathbf{x}_A)$$

$$\begin{aligned}
 \mathbf{F}(\lambda_C, \theta_C, \rho_C) \begin{bmatrix} \widehat{\lambda}_C - \lambda_C \\ \widehat{\theta}_C - \theta_C \\ \widehat{\rho}_C - \rho_C \end{bmatrix} &\approx \mathbf{A} \left(\mathbf{F}(\lambda_B, \theta_B, \rho_B) \begin{bmatrix} \widehat{\lambda}_B - \lambda_B \\ \widehat{\theta}_B - \theta_B \\ \widehat{\rho}_B - \rho_B \end{bmatrix} \right. \\
 &\left. - \mathbf{F}(\lambda_A, \theta_A, \rho_A) \begin{bmatrix} \widehat{\lambda}_A - \lambda_A \\ \widehat{\theta}_A - \theta_A \\ \widehat{\rho}_A - \rho_A \end{bmatrix} \right).
 \end{aligned}$$

It follows that

$$\begin{aligned}
 \widehat{\mathbf{Cov}} \left(\begin{bmatrix} \widehat{\lambda}_C - \lambda_C \\ \widehat{\theta}_C - \theta_C \\ \widehat{\rho}_C - \rho_C \end{bmatrix} \right) &= \mathbf{G} \widehat{\mathbf{Cov}} \left(\begin{bmatrix} \widehat{\lambda}_B - \lambda_B \\ \widehat{\theta}_B - \theta_B \\ \widehat{\rho}_B - \rho_B \end{bmatrix} \right) \mathbf{G}^T \\
 &+ \mathbf{H} \widehat{\mathbf{Cov}} \left(\begin{bmatrix} \widehat{\lambda}_A - \lambda_A \\ \widehat{\theta}_A - \theta_A \\ \widehat{\rho}_A - \rho_A \end{bmatrix} \right) \mathbf{H}^T
 \end{aligned} \tag{53.43}$$

where

$$\mathbf{G} = \mathbf{F}(\widehat{\lambda}_C, \widehat{\theta}_C, \widehat{\rho}_C)^{-1} \hat{\mathbf{A}} \mathbf{F}(\widehat{\lambda}_B, \widehat{\theta}_B, \widehat{\rho}_B)$$

$$\mathbf{H} = \mathbf{F}(\widehat{\lambda}_C, \widehat{\theta}_C, \widehat{\rho}_C)^{-1} \hat{\mathbf{A}} \mathbf{F}(\widehat{\lambda}_A, \widehat{\theta}_A, \widehat{\rho}_A) \tag{53.44}$$

(53.42)–(53.44) constitute the analogy to (53.35) when the matrix exponential parameterization (53.23) of $\text{SO}(3)$ is replaced by axis latitude, longitude, and angle of rotation.

The Map Maker’s Dilemma (Revisited)

As discussed near equation (53.5), and similar approximations (53.9) and (53.10), when using $\mathbf{e} \in (\mathbf{Au})^\perp$ to represent the deviation $\mathbf{v} \in S^2$ from its “true value” $\mathbf{Au} \in S^2$, we are essentially using a polar projection, centered at \mathbf{Au} , $R^2 \approx (\mathbf{Au})^\perp \rightarrow S^2$. Any map from $R^2 \rightarrow S^2$ is distorting somewhere, but when \mathbf{v} is close to \mathbf{Au} , the distortions in this polar projection will be small.

A parameterization of S^2 by two parameters is essentially a map $R^2 \rightarrow S^2$. The commonly used map (latitude, longitude) is highly distorting near the north and south poles and hence would not be appropriate to represent \mathbf{v} when \mathbf{Au} has a latitude much different from 0° .

Thinking temporarily of the unit quaternions S^3 instead of $\text{SO}(3)$, equation (53.39) represents a point $q_{\hat{A}} \in S^3$, that is, close to $q_A \in S^3$ by a tangent vector $q_A \tilde{x}_A \in q_A^\perp$. Essentially we are mapping S^3 using a polar projection centered at q_A . Hopefully $q_{\hat{A}}$ will be close to q_A and the distortions introduced by the polar projection will be small. Indeed these distortions are asymptotically trivial.

Because of the close relationship of S^3 to $\text{SO}(3)$, the same observations apply to using h_A to represent $\hat{\mathbf{A}}$ as in $\hat{\mathbf{A}} = \mathbf{A}\Phi(h_A)$.

The map $R^3 \rightarrow \text{SO}(3)$ using axis latitude, longitude, and angle of rotation also introduces distortions and these distortions can certainly exist in the area of interest, that is, rotations close to \mathbf{A} . (53.42)–(53.44) are considerably more complicated than (53.35) and this is due to the unfortunate matrices $\mathbf{F}(\lambda, \theta, \rho)$ whose sole functions are to correct the distortions introduced by an inappropriate parameterization of $\text{SO}(3)$.

References

1. Chang, T.: Image registration, rigid bodies, and unknown coordinate systems. In: Pham, H. (ed.) *Springer Handbook of Engineering Statistics*, pp. 571–590. Springer-Verlag, London (2006)
2. Chang, T.: On the statistical properties of estimated rotations. *J. Geophys. Res.* **92**(B7), 6319–6329 (1987)
3. Chang, T.: Spherical regression and the statistics of tectonic plate reconstructions. *Int. Stat. Rev.* **61**, 299–316 (1993)
4. Chang, T.: Spherical regression. *Ann. Stat.* **14**(3), 907–924 (1986)
5. Rivest, L.-P.: Spherical regression for concentrated Fisher-von Mises distributions. *Ann. Stat.* **17**(1), 307–317 (1989)
6. Chang, T., Ko, D.: M-estimates of rigid body motion on the sphere and in Euclidean space. *Ann. Stat.* **23**(5), 1823–1847 (1995)
7. Rivest, L.-P.: A correction for axis misalignment in the joint angle curves representing knee movement in gain analysis. *J. Biomech.* **38**, 1604–1611 (2005)
8. Oualkacha, K., Rivest, L.-P.: On the estimation of an average rigid body motion. *Biometrika.* **99**, 585–598 (2012)
9. Chang, T., Rivest, L.-P.: M-estimation for location and regression parameters in group models: a case study using Stiefel manifolds. *Ann. Stat.* **29**(3), 784–814 (2001)

10. Mardia, K.V., Patrangenaru, V.: Directions and projective shapes. *Ann. Stat.* **33**(4), 1666–1699 (2005)
11. Chang, T., Stock, J., Molnar, P.: The rotation group in plate tectonics and the representation of uncertainties of plate reconstructions. *Geophys. J. Int.* **101**, 649–661 (1990)
12. Stock, J., Molnar, P.: Some geometrical aspects of uncertainties in combined plate reconstructions. *Geology*. **11**, 697–701 (1983)
13. Molnar, P., Stock, J.: A method for bounding uncertainties in combined plate reconstructions. *J. Geophys. Res.* **90**, 12,537–12,544 (1985)
14. Chang, T.: Estimating the relative rotation of two tectonic plates from boundary crossings. *J. Am. Stat. Assoc.* **83**(404), 1178–1183 (1988)
15. Muller, R.D., Sdrolias, M., Gaina, C., Roest, W.R.: Age, spreading rates and spreading symmetry of the world's ocean crust. *Geochem. Geophys. Geosyst.* **9**, Q04006 (2008). <https://doi.org/10.1029/2007GC001743>
16. Hellinger, S.J.: The uncertainties of finite rotations in plate tectonics. *J. Geophys. Res.* **86**, 9312–9318 (1981)
17. Kirkwood, B.H., Royer, J.-Y., Chang, T., Gordon, R.G.: Statistical tools for estimating and combining finite rotations and their uncertainties. *Geophys. J. Int.* **137**, 408–428 (1999)
18. Royer, J.-Y., Chang, T.: Evidence for relative motions between the Indian and Australian plates during the last 20 Myr from plate tectonic reconstructions. Implications for the deformation of the Indo-Australian plate. *J. Geophys. Res.* **96B**, 11779–11802 (1991)
19. DeMets, C., Gordon, R.G., Royer, J.-Y.: Motion between the Indian, Capricorn, and Somalian plates since 20 Ma: implications for the timing and magnitude of distributed lithospheric deformation in the equatorial Indian ocean. *Geophys. J. Int.* **161**, 445–468 (2005)
20. Kirkwood, B.H., Chang, T.: Combining estimates of tectonic plate rotations: an extension of Welch's method to spherical regression. *J. Mult. Anal.* **65**, 71–108 (1998)
21. Johansen, S.: The Welch-James approximation to the distribution of the residual sum of squares in a weighted linear regression. *Biometrika*. **67**, 85–92 (1980)
22. Goldstein, H.: *Classical Mechanics*. Addison-Wesley, Reading Massachusetts, Palo Alto, London (1951)



Ted Chang received his Ph.D. in mathematics (algebraic topology) from the University of California, Berkeley in 1972. He is currently an Emeritus Professor of Statistics at the University of Virginia. His major fields of research are the statistics of rigid body motion, with an emphasis on tectonic plate reconstructions, and the analysis of finite population sample surveys.



Statistical Modeling of Discrete Choices for Human Behaviors

54

Xi Zhu and Shuai Huang

Contents

54.1	Introduction	1075
54.2	The Problem of Discrete Choice Modeling	1076
54.3	Random Utility Maximization (RUM) Theory	1077
54.3.1	Binomial Choice Model.....	1077
54.3.2	Multinomial Logit Model.....	1078
54.4	The Random Regret Minimization (RRM) Model	1079
54.4.1	RRM Model Specification.....	1079
54.4.2	Some Variants of RRM.....	1080
54.5	The Relative Advantage Maximization (RAM) Model	1080
54.6	Extensions to More Complicated Choice Sets	1081
54.6.1	Nested Logit Model.....	1081
54.6.2	Mixed Logit Model.....	1082
54.7	Numerical Study	1083
54.7.1	General Description.....	1083
54.8	Discussion	1084
54.8.1	Limitations and Challenges.....	1084
54.8.2	Irrational Behavior.....	1085
References	1085

start from the random utility maximization (RUM) theory and present its basic usage in binary and multinomial choice scenarios. We also present the random regret minimization (RRM) theory and relative advantage maximization (RAM) theory. Some extensions of the RUM are also presented, including the nested logit model that allows dependencies among alternatives and mixed logit model that considers the individual heterogeneity. We illustrate the usage and interpretation of these models through a numeric case study.

Keywords

Discrete choice models · Random utility maximization (RUM) · Random regret minimization (RRM) · Relative advantage maximization (RAM)

Abstract

Human behavior models, especially the choice models, have been developed and widely used in both research and practice for decades. They help explain human behaviors by modeling the choices made by each individual. Successful applications of these models require good understandings of the properties and the assumptions of the models. This chapter summarizes some commonly used discrete choice models in engineering area. Different assumptions toward the mechanism of choice-making behaviors lead to different types of choice models. We

54.1 Introduction

Human behavior modeling has drawn increasing attention for its potential in various realms such as economics, marketing, and engineering. A better understanding on individual preferences and human choice-making behavior may lead to more satisfactory, targeted, and efficient services. Because of this, researchers have never stopped trying to develop models to explain and predict human behavior in different circumstances in the last decades [21, 33, 34]. In this chapter, we focus on the choice-making behaviors that predominate many kinds of human behaviors according to Glasser's choice theory psychology [22]. Many models have been developed and implemented to model choice-making behaviors with different assumptions. One of the most widely accepted models, the discrete choice modeling, is used to describe and predict the probability of selecting a choice from a finite choice set in a probabilistic fashion, based on the Random Utility Maximization (RUM) principle [28]. RUM model requires a preference order over all choices of the finite

X. Zhu · S. Huang (✉)
University of Washington, Seattle, WA, USA
e-mail: zhux2012@uw.edu; shuaih@uw.edu

alternative set, which means that an axiom of “transitivity” should be satisfied. This means, for example, if alternative A is weakly preferred to alternative B, and B to C, then A is weakly preferred to C. If the preference order fails to meet the requirement of “transitivity,” the utility function in RUM model may not exist, and the choice behavior cannot be formulated following the RUM principle [32].

This specification to represent the utility of an alternative is usually called “context-independent” in studies, assuming that the utility of an alternative is invariant no matter whether other alternatives are present or not [4, 8]. However, this may not be the case when individuals are making choices. Imagine that Susan is considering going for a walk around a lake which is about 2 miles away from her home. She may prefer driving to the lake comparing with taking public transit there. When the two alternatives are taking public transit and biking, she may choose to take the public transit. However, when she is asked to either drive or bike, she would choose to bike. These kinds of non-transitive, circular orderings are pretty common in reality, which means that the traditional RUM model may not be applicable here [32].

Therefore, “context-dependent” models, such as RAM and RRM model where the transitivity on an alternative set may not be bounded, arise. The basic idea of RAM, the relative advantage maximization model, is to add the relative advantage of an alternative into its utility function as a context-dependent component to be maximized [26, 44]. The main hypothesis of RRM, the random regret minimization model, is that people are seeking to minimize the negative emotions when making decisions [14, 16, 26]. One may also regard the context-dependent specification as that people are not absolutely rational but with bounded rationality. This is because that people are bounded by the time and other resources required for their utilization in the face of the urgency and importance of the particular choice problem at hand. Several studies have shown the success of these models in applications.

The purpose of this chapter is to review the important models and techniques that can be used to model and predict user’s choice-making behaviors. For example, with these models, personalized travel choice recommendation systems can promote alternatives that are more likely to be accepted by an individual, helping him/her make plans for travel or triggering desired travel behavioral changes. We will also summarize properties and limitations of each model. This chapter is organized as follows: we start from traditional discrete choice model with RUM specification, including binary logit model, multinomial logit model, nested logit model, and mixed logit model. Then the RRM model is presented, followed by the RAM model. After that, we present a numerical study, in which we apply three models to a single dataset and illustrate how they compare with each other. Toward the end of the chapter, we will also discuss some

other models, such as machine learning methods, the hidden Markov model, and Bayesian statistics that have been useful in recommendation systems.

54.2 The Problem of Discrete Choice Modeling

Transportation is also an area in which discrete choice models are commonly used, since many travel-related choices such as travel mode and destination are discrete, qualitative, and finite [11, 39].

For discrete choice model, there are several requirements for the choice set [39]:

1. The choice set must be a finite set, which means there should be a finite number of alternatives.
2. The choice set includes all possible alternatives the system can offer to a user, and the user will choose an alternative from the set.
3. The alternatives in the choice set should be mutually exclusive, and the user chooses only one alternative from the set.

Discrete choice model considers that an individual’s choice behavior among a finite set of alternatives is influenced by a number of factors, including some socioeconomic and sociodemographic characteristics of the individual, the attributes of the alternatives in the choice set, and the external environments. These factors, together with a set of parameters quantifying the effects of these factors, collectively contribute to the utility that the individual obtains from choosing an alternative.

However, it is practically impossible to include all factors that could impact an individual’s choice in the utility function, since they are probably only partially observed, and it is unlikely that we could measure every factor perfectly and accurately. To count for these unobserved and immeasurable effects, the discrete choice model incorporates stochastic specifications, i.e., random utility terms. Possible causes of the randomness in user behaviors include the unobserved factors of the choices, the preference variation over people (heterogeneity), and correlation among choices in choice sets.

A systematic characterization and notations of the discrete choice problem are presented in Table 54.1. Given the depth of this problem, a range of models have been developed. For example, it is common that the utility function in discrete choice model is expressed in a linear form [9, 39]:

$$U_{ij} = \mathbf{x}_{ij}\boldsymbol{\beta}_i + \epsilon_{ij} = \mathbf{V}_{ij} + \epsilon_{ij} \quad (54.1)$$

where \mathbf{x}_{ij} is a vector of all factors from different categories (as mentioned above) of alternative j for individual i , $\boldsymbol{\beta}_i$ is a

Table 54.1 Table of notations

Notation	Meanings
i	Individual i
j, k	Alternative j , alternative k
m	m th attribute of an alternative
J	Total number of alternatives in a choice set
U_{ij}	Utility of alternative j for individual i
\mathbf{x}_{ij}	Vector of attributes of alternative j for individual i
$\boldsymbol{\beta}_i$	Vector of parameters/preferences of individual i corresponding to all attributes
ϵ	Random term drawn from a certain distribution
V_{ij}	Systematic utility of alternative j for individual i
P_{ij}	Probability for individual i choosing alternative j
y_0	The choice made to alternative 0 in a binary choice. If $y_0 = 1$, the individual accept alternative 0; if not, $y_0 = 0$
ΔV_{01}	Difference between V_0 and V_1
$\mathbf{x}_{1,0}$	Vector of attribute differences between \mathbf{x}_1 and \mathbf{x}_0
$G(\cdot), G(\boldsymbol{\beta} \boldsymbol{\theta})$	A certain type of distribution, with $\boldsymbol{\theta}$ as a set of parameters that could describe the distribution

Notations used in the chapter

vector of parameters that quantify the effects of these factors, and ϵ_{ij} is the random utility term. Here, V_{ij} is the systematic utility, and ϵ_{ij} is the random utility. In what follows, we will introduce a range of models, including the RUM, RAM, and RRM models.

54.3 Random Utility Maximization (RUM) Theory

According to the random utility maximization (RUM) model, individual i chooses the alternative j over all other alternatives because that choice provides the highest utility/benefits she/he could obtain, reflecting a rational behavior of the indi-

vidual. In other words, the probability for him/her to select alternative j over alternative k from a set J is [11, 39]:

$$\begin{aligned} P_{ij} &\equiv P(\text{Individual } i \text{ choosing alternative } j) \\ &= P(U_{ij} \geq U_{ik}) \quad \forall j \neq k \in (1, 2, \dots, J). \end{aligned} \quad (54.2)$$

This is the overall framework, and in what follows we introduce two specific models that expand on this premise.

54.3.1 Binomial Choice Model

Binomial choice model is for circumstances when there are only two alternatives (denoted as alternative 0 and 1). The logit model and probit model are two common examples of binomial choice model. The two models have different assumptions on the random term in Eq. 54.1.

Logit model assumes that the random term ϵ_j in Eq. 54.1 is an i.i.d. (independent and identically distributed) extreme value that follows a Gumbel distribution [39]. Suppose that the utilities of the two alternatives are U_0 and U_1 , respectively. By substituting Eq. 54.1 into Eq. 54.2, we could have the following equation:

$$\begin{aligned} P_0 &= P(U_0 \geq U_1) \\ &= P(V_0 + \epsilon_0 \geq V_1 + \epsilon_1) \\ &= P(V_0 - V_1 \geq \epsilon_1 - \epsilon_0). \end{aligned} \quad (54.3)$$

Since the logit model assumes that the $\epsilon_0 \sim \text{Gumbel}(a_0, b)$ and $\epsilon_1 \sim \text{Gumbel}(a_1, b)$, their difference $\epsilon_1 - \epsilon_0 \sim \text{Logistic}(a_1 - a_0, b)$ follows a logistic distribution [39]. From Eq. 54.3 we could see that the probability for an individual to choose alternative 0 depends only on the difference in utilities between the two alternatives, not on the absolute level of utilities. Thus, we further derive that

$$\begin{aligned} P_0 &= P(V_0 - V_1 \geq \epsilon_1 - \epsilon_0) \\ &= P(\Delta V_{01} \geq \epsilon) \quad (\text{Let } V_0 - V_1 = \Delta V_{01}, \text{ and } \epsilon_1 - \epsilon_0 = \epsilon) \\ &= P(\epsilon \leq \Delta V_{01}) \\ &= \text{logit}^{-1}(\Delta V_{01}) \end{aligned} \quad (54.4)$$

Different from the logic model, the probit model assumes that the random term ϵ_j in Eq. 54.1 follows a standard normal distribution. The two curves shown in Fig. 54.1 represent the two cumulative distribution functions of the two types of the random utility term.

It can be seen that using the Gumbel distribution for the errors (and hence the logistic distribution for the error

differences) is nearly the same as assuming that the errors are independently normal [39]. The difference is that the two tails of the Gumbel distribution are slightly fatter than those of a normal distribution, which means that the Gumbel distribution allows more aberrant behaviors than the normal one. However, the difference is usually indistinguishable empirically [39].

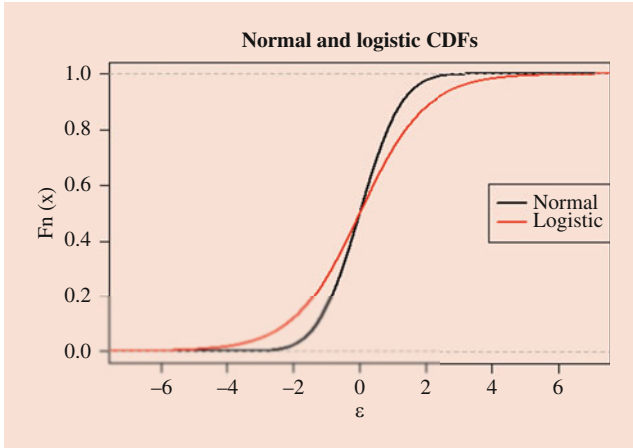


Fig. 54.1 CDFs for norm distribution and logistic distribution

With logistic model specification, the probability for an individual to choose alternative 0 is

$$P_0 = P(y_0 = 1) = \frac{\exp(V_0)}{\exp(V_0) + \exp(V_1)} = \frac{1}{1 + \exp(V_1 - V_0)} \tag{54.5}$$

In the above equation, $V_0 = \beta_0 \mathbf{x}_0$ and $V_1 = \beta_1 \mathbf{x}_1$ are the constant parts of the utilities of the two alternatives. From the equation we could have another property of the discrete choice model: the sum of the probability of all alternatives will equal 1, i.e., $P(y_0 = 1) + P(y_1 = 1) = 1$.

To interpret the estimates of the model, we need to talk about the concept of odds. This term is used to describe the probability of an event (or the probability of a nonevent). For the binary choice model, the odds are

$$\begin{aligned} Odds_{y_0=1} &= \frac{P(y_0 = 1)}{P(y_0 = 0)} = \frac{P(y_0 = 1)}{1 - P(y_0 = 1)} \\ &= \exp(V_1 - V_0) = \exp(\mathbf{x}_{1,0} \boldsymbol{\beta}) \end{aligned} \tag{54.6}$$

where $\mathbf{x}_{1,0}$ is the vector of attribute differences between \mathbf{x}_1 and \mathbf{x}_0 . Take log on both sides; the equation could further be written as

$$\ln(Odds_{y_0=1}) = \mathbf{x}_{1,0} \boldsymbol{\beta} \tag{54.7}$$

We could see the coefficients in a simple logistic model in terms of the log odds. For example, the coefficient β_a in $\boldsymbol{\beta}$ implies that a one-unit change in x_a in $\mathbf{x}_{1,0}$ results in a β_a -unit change in the log of the odds.

The estimation of the binomial logit model uses the maximum likelihood estimation (MLE), with which the estimated parameters for $\boldsymbol{\beta}$ have the greatest likelihood/probability to obtain the observed choices.

We use an example to illustrate the estimation method. Assuming that we have M observed binary choice behaviors $(y_1, \mathbf{x}_1), (y_2, \mathbf{x}_2), \dots, (y_M, \mathbf{x}_M)$ for an individual, we would like to predict his/her choice-making behavior with a binary logit model. For each observation m , assume that $P(y_m = 1) = \frac{\exp(\mathbf{x}_m \boldsymbol{\beta})}{1 + \exp(\mathbf{x}_m \boldsymbol{\beta})}$, and $P(y_m = 0) = 1 - \frac{\exp(\mathbf{x}_m \boldsymbol{\beta})}{1 + \exp(\mathbf{x}_m \boldsymbol{\beta})}$. The likelihood function could be formulated as

$$L(y|\mathbf{x}; \boldsymbol{\beta}) = \prod_{m \in \mathcal{M}_+} \frac{\exp(\mathbf{x}_m \boldsymbol{\beta})}{1 + \exp(\mathbf{x}_m \boldsymbol{\beta})} \prod_{m \in \mathcal{M}_-} \left[1 - \frac{\exp(\mathbf{x}_m \boldsymbol{\beta})}{1 + \exp(\mathbf{x}_m \boldsymbol{\beta})} \right] \tag{54.8}$$

where \mathcal{M}^+ refers to the observations for which $y_m = 1$ and \mathcal{M}^- to the observations for which $y_m = 0$.

The log likelihood function for the example is

$$\begin{aligned} \ln L(y|\mathbf{x}; \boldsymbol{\beta}) &= \sum_{m=1}^M \left\{ y_m \ln \frac{\exp(\mathbf{x}_m \boldsymbol{\beta})}{1 + \exp(\mathbf{x}_m \boldsymbol{\beta})} + (1 - y_m) \left[1 - \ln \frac{\exp(\mathbf{x}_m \boldsymbol{\beta})}{1 + \exp(\mathbf{x}_m \boldsymbol{\beta})} \right] \right\} \\ &= \sum_{m=1}^M \{ y_m [\mathbf{x}_m \boldsymbol{\beta} - \ln(1 + \exp(\mathbf{x}_m \boldsymbol{\beta}))] - (1 - y_m) \ln(1 + \exp(\mathbf{x}_m \boldsymbol{\beta})) \} \end{aligned} \tag{54.9}$$

The MLE of $\boldsymbol{\beta}$ maximizes the log likelihood function of Eq. 54.9.

54.3.2 Multinomial Logit Model

When there are J alternatives ($J > 2$) in the choice set, the multinomial logit model (MNL) casts the problem as a set of $J - 1$ independent binomial choice problems. A “reference” alternative is assigned before the regression such that other $J - 1$ alternatives can compare against it, one at a time. The same applies to the binomial logit model, as one of the

alternative is “reference” and the other is comparing against it. In MNL, any one of the alternative in the choice set could be set as the reference. However, if a reference alternative is decided, it may not be changed in the whole modeling process. Notice that though different reference alternative may return different estimates of the parameter vector $\boldsymbol{\beta}$, it is shown that the models are mathematically equivalent.

Similar to binary choice model, the choice probability for individual i to choose alternative k is formulated as

$$P_k = P(y_k = 1) = \frac{\exp(V_k)}{\sum_{j \in \{1,2,\dots,J\}} \exp(V_j)} \tag{54.10}$$

For a multinomial logit model, a set of parameters will be estimated for each alternative j comparing with the reference alternative. The interpretation of the parameters estimated is the same as the interpretation of the parameters in a binary logit model, where the two alternatives are the alternative j and the reference.

54.4 The Random Regret Minimization (RRM) Model

Random regret minimization (RRM) model differs from RUM model in terms of the decision-making principle. Rather than assuming that an individual is choosing the alternative that could provide the highest utility/benefit, the main hypothesis in RRM model is that the individual will choose the alternative such that the “emotion of regret” associated with the choice is minimized [13–16]. The “emotion of regret” refers to the negative emotion of loss brought by the non-chosen alternatives. This might happen when some attributes of the chosen alternative are worse than those of the non-chosen alternatives. For example, Susan may choose to drive to the lake finally. Though she believes that driving is the best choice among the three alternatives she is considering (drive, take public transit, bike), she still has a feeling of loss when considering that the cost of biking is much lower than driving or biking is healthier for herself. The “emotion of regret” is related to the advantages of the non-chosen alternatives. Here, an alternative has “advantage” on an attribute means that, regardless of other attributes, the attribute of this alternative is more attractive comparing with the same attribute of another alternative.

Different from an RUM model where transitivity is required for the utility of all the alternatives, here the “regret” of an alternative in an RRM model makes references to the values of the same attribute of other alternatives presented for comparison. This means that “regret” is not a concept with transitivity (remember that the concept of “utility” in the RUM model has transitivity) and can be different when the presented alternatives change.

The RRM model can be used to analyze and model a wide variety of choice behaviors. For example, in transportation, it has been applied to departure time choices, route choices, mode-destination choices, etc. Though RUM model may also work in these scenarios, RRM model may be able to handle some other problems that could not be explained by RUM model. A well-known example is the “compromise effect” that could be observed in human decision-making process: individuals tend to choose an alternative that is moderately good on all factors, avoiding extreme alternatives (including both extremely good or extremely poor) [13, 25]. This seemingly irrational behavior could not be well captured by

RUM models but could be reasonably interpreted by an RRM model.

Since the two types of model hold different assumptions toward the underlying mechanism of choice-making behavior, they provide valuable perspectives and implications. Because of this, RRM model may serve as a supplement to individual choice behavior modeling toolbox.

54.4.1 RRM Model Specification

According to [12], it is hypothesized that the regret associated with alternative k , due to a comparison with a competing alternative j based on a particular attribute m , equals zero in case alternative k scores equal or better than j on attribute m . Otherwise, the regret associated with the attribute comparison is a nondecreasing function of the difference in attribute value.

For example, assume that individual i is facing a choice among alternatives j , k , and l . The attributes related to the choice-making process is x_m and x_n : $j = \{x_{mj}, x_{nj}\}$, $k = \{x_{mk}, x_{nk}\}$, $l = \{x_{ml}, x_{nl}\}$. Each alternative is associated with a measure of random regret RR_j , RR_k , and RR_l . When individual i is making choices, he/she minimizes this random regret, which means that he/she chooses the choice $\arg_v \min_{v=j,k,l} \{RR_v\}$.

The classical RRM model proposed by Chorus et al. is given by the following equation [12, 14, 16]:

$$RR_{ki} = \sum_{j \neq k} \sum_m \ln(1 + \exp(\beta_m [x_{jmi} - x_{kmi}])) + \epsilon_{ki} \quad (54.11)$$

where i represents individual i , m represents the m th attribute, and j represents an competing alternative j .

Equation 54.11 shows that for individual i , when he/she considers selecting alternative k , the “regret” is the sum of all binary regrets associated with the binary comparisons between alternative k and its competitor alternatives j ($j \neq k$). For each binary comparison, the binary regrets are the sum of the differences between all attributes m of an alternative j and the alternative k .

The authors who propose RRM model also show that [13]:

1. The marginal regret converges to β_m when $x_{jm} - x_{km}$ becomes sufficiently large. The overall regret is increasing with the increasing number of attributes on which alternative k is outperformed, as well as with the increasing number of alternatives by which alternative k is outperformed and the increasing importance of the attribute β_m .
2. The marginal regret with respect to attribute m when considering alternative k approaches zero when $x_{jm} - x_{km} < 0$, i.e., when the chosen alternative k outperforms

the competing alternative j ($j \neq k$). Hence, the RRM model postulates that when a decision-maker considers alternative k as compared to alternative j he or she experiences (almost) no regret with regard to attribute m when in alternative k the m 's attribute performs considerably better. This is consistent with our intuition that regrets may arise only when alternative k is outperformed by the competing alternative j on attribute m .

The random error term ϵ_{ki} in Eq. 54.11 is assumed to be i.i.d. type I extreme value similar to RUM model. By letting the variance of the Gumbel distribution be $\pi^2/6$, the error term could be “integrated out,” and the system regret (no random term) is

$$R_{ki} = \sum_{j \neq k} \sum_m \ln(1 + \exp(\beta_m [x_{jmi} - x_{kmi}])) \quad (54.12)$$

Referring to [13, 14, 16], since minimization of random regret is equivalent to maximization of minus random regret, and given the symmetrical nature of the i.i.d. errors, the choice probabilities can be taken on the well-known MNL form. Thus, with Eq. 54.12, the choice probability associated with alternative k equals

$$P_{ki} = \frac{\exp(-R_{ki})}{\sum_{j=1,2,\dots,J} \exp(-R_{ji})} \quad (54.13)$$

Similar to RUM model, the parameters of RRM model can also be estimated with its likelihood function, given its properties of the smoothness, differentiability, and globally concaveness. A book on RRM and related tutorials can be found in [13], and some sample codes can also be found in [40].

54.4.2 Some Variants of RRM

Some variants of the RRM model have also been developed, such as the μ RRM and PRRM models. These models can be supplements of the classical RRM model [8, 41].

The μ RRM model is a generalization of the classical RRM model. Since the size of the estimated taste parameters (β s) may significantly impact the extent to which losses (regrets) loom larger than equivalent gains (rejoices), the μ RRM model accommodates for different shapes of the attribute level regret functions by adding a scale parameter μ to adjust the size of the estimated taste parameters:

$$RR_k^{\mu RRM} = \sum_{j \neq k} \sum_m \ln \left(1 + \exp \left(\frac{\beta_m}{\mu} [x_{jm} - x_{km}] \right) \right) + \epsilon_k$$

$$- \epsilon_{ki} \sim i.i.d. \text{ EV}(0, \mu) \quad (54.14)$$

Notice that the scale parameter μ is also to be estimated. When μ is estimated to be 1, then the μ RRM becomes the classical RRM. When μ is close to zero, the PRRM model is obtained, which postulates the strongest regret minimization behavior which is possible within the RRM modeling paradigm [8, 41].

54.5 The Relative Advantage Maximization (RAM) Model

Relative advantage maximization model, abbreviated as “RAM” model, is another context-dependent model that could be used to describe and predict user choice behaviors and is able to explain the “compromise effect” [26]. Similar to RRM model, RAM model also compares each attribute of the chosen alternative with the same attribute of the competing alternatives.

In RAM model, the “disadvantage” of an individual i 's choosing alternative k over a competing alternative j with respect to attribute m is equivalent to the concept of “regret” in RRM model. Thus, denoted by D_{mkji} , the disadvantage is formulated in the same form as the RRM model:

$$D_{mkji} = \ln(1 + \exp[\beta_m (x_{mji} - x_{mki})]) \quad (54.15)$$

The key difference between RAM model and RRM model is that RAM model takes both the “disadvantage” and “advantage” of an alternative into account. The “disadvantage” and “advantage” are assumed to be symmetric in RAM model [26]. This means that the advantage for an individual i to choose alternative k over j with respect to attribute m (denoted as A_{mkji}) equals the corresponding disadvantage for individual i to choose alternative j over k with respect to the same attribute m :

$$A_{mkji} = D_{mjki} = \ln(1 + \exp[\beta_m (x_{mji} - x_{mki})]) \quad (54.16)$$

Notice that the subscripts of A_{mkji} and D_{mjki} in Eq. 54.16 is different. The disadvantage D_{mkji} for an individual i to choose alternative k over j with respect to attribute m corresponding to A_{mkji} can be written as $D_{mkji} = \ln(1 + \exp[\beta_m (x_{mki} - x_{mji})])$. Please notice the difference between this equation and Eq. 54.16.

Also, Eq. 54.16 is only for a single attribute m . We could add up all the attributes, and the total advantage and disadvantage for individual i to choose alternative k over j over all attributes are as follows [8]:

$$A_{kji} = \sum_m A_{mkji}$$

$$D_{kji} = \sum_m D_{mkji} \quad (54.17)$$

RAM model defines the term “relative advantage” (RA) of alternative k over competing alternative j as a ratio of k 's advantage (over j) over the sum of advantages and disadvantages (over j). Mathematically, the relative advantage of the individual i to choose alternative k over j is:

$$RA_{kji} = \frac{A_{kji}}{A_{kji} + D_{kji}} \quad (54.18)$$

RAM model also integrates the definition of the systematic utility in MNL together with its concept of relative advantage, such that both context-independent preferences and context-dependent preferences play roles in the model. The systematic utility for individual i to choose an alternative k could then be written as a linear combination of the systematic utility term in MNL and the relative advantage term defined in previous:

$$V_{ki}^{RAM} = \beta_m + \sum_m \beta_m x_{mki} + \sum_{j \neq k} RA_{kji} \quad (54.19)$$

Here x_{mki} is the context-independent attribute m for individual i to choose an alternative k .

54.6 Extensions to More Complicated Choice Sets

In this section, we mention a few models that illustrate some interesting ideas to tackle complicated choice sets, i.e., when the assumptions about the choice set presented in Sect. 54.2 are not met. The discrete choice models we have discussed so far could address many choice behavior problems in applications. However, some limitations exist. For instance, the model may fail to work when alternatives are not independent from each other. One assumption for RUM models is that there is no correlation in unobserved factors over alternatives, which is known as the “independence of irrelevant alternatives” (IIA) property. In other words, IIA assumption says that when individuals are making choices among a set of alternatives, the probabilities for them to choose A over B should not be impacted by whether another alternative C is present or not [2,5,29]. It could be easily noticed that this lack of correlation among alternatives are not always be realistic. Thus, some models, such as nested logit model which allow correlation over alternatives in a certain way, were developed. Another problem is that the standard logit model's “taste” coefficients, i.e., β 's, are fixed, which means the β 's are the same for everyone. To allow each individual to have different β , mixed logit model, which assumes that the β is a random variable following a certain distribution, was developed. In what follows, we discuss two models, the nested logit model, and the mixed logit model, where both are extensions of the RUM model.

54.6.1 Nested Logit Model

Nested logit model is used to deal with choice problems where there are nonzero correlations across alternatives, i.e., the IIA is violated [42]. In other words, nested logit model is based on the idea that some alternatives may be joined in several groups (called nests). The error terms may present some correlation in the same nest, whereas error terms of different nests are still uncorrelated. In this way, nested logit model allows for nonzero correlation between unobserved components of choices within a nest and maintains zero correlation across nests.

After a nested logit model groups together alternatives that are similar into a nest, each nest with multiple alternatives in turn is considered as a new composite alternative, which competes with the rest of alternatives (other alternatives that are not in the nests and other nests) available to the individual. Notice that for a nested logit model, there can be several nests in a choice set, and each alternative can be in no more than one nest.

Whether to use nested logit model and how to group alternatives if necessary may require an IIA test. For example, the IIA test in R software tests the probability ratio of any two alternatives, seeing whether the results obtained on the estimation with all the alternatives and only on a subset of the alternatives are consistent. Illustrating this in a mathematical way: assume that there are N alternatives in a choice set \mathcal{C} . \mathcal{C}' is subset of \mathcal{C} , which only includes two alternatives j and k . The IIA test in R is to see whether $\frac{P(y=k, k \in \mathcal{C})}{P(y=j, j \in \mathcal{C})}$ equals to $\frac{P(y=k, k \in \mathcal{C}')}{P(y=j, j \in \mathcal{C}')}$.

Let's use an example to explain the specification of the nested logit model, as shown in Fig. 54.2: there are three alternatives available at the beginning, car, light rail, and bus. If there is no other conditions and we simply use MNL, we may have a conclusion that each of the alternatives has the same probability to be chosen (33.3%). However, this may

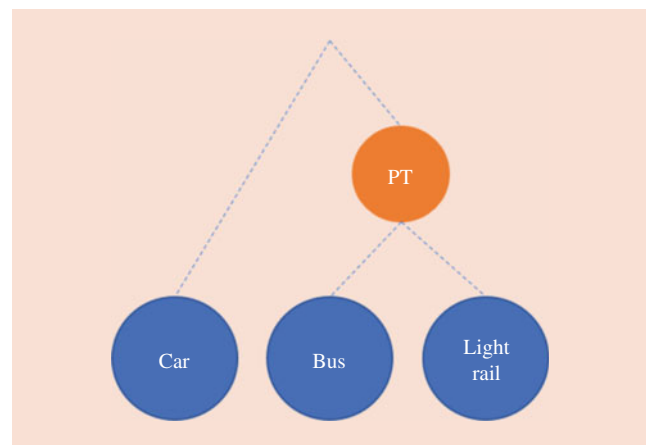


Fig. 54.2 Nested alternative

not be the case as the bus and light rail are both public transit (PT) and can be correlated. Nested logit model groups these two alternatives into a composite alternative PT, and then the available alternatives become Car and PT.

With nests identified, nested logit model estimates the probabilities of each alternative hierarchically:

1. Lower level: estimate a multinomial logit model for the alternatives in each nest.
2. Higher level: estimate a multinomial logit model consisting of all composite alternatives and alternatives which are non-nested.

To use a composite alternative to represent all the alternatives in the nest when building the higher level MNL, the utility of the composite alternative needs to include two components: (1) the expected maximum utility (EMU) of the lower level alternatives in the nest and (2) the vector of attributes which are common to all members of the nest. The probability that an alternative in a nest is chosen is computed as the product of the marginal probability of choosing the composite alternative and the conditional probability of choosing the alternative among all alternatives in the nest.

In the example shown in Fig. 54.2, bus and light rail are nested and a composite alternative PT is generated. Then the lower level, the PT nest, would be modeled by a binary logit model:

$$P(y_{bus} = 1|PT) = \frac{\exp(W_{bus})}{\exp(W_{rail}) + \exp(W_{bus})} \quad (54.20)$$

$$P(y_{rail} = 1|PT) = 1 - P(y_{bus} = 1)$$

where the utilities W_{bus} and W_{rail} contain only those attributes which are not the same to both alternatives.

The higher level would be modeled by another binary logit model:

$$P(y_j = 1) = \frac{\exp(V_j)}{\exp(V_{car}) + \exp(V_{PT})} \quad j \in (car, PT) \quad (54.21)$$

In Eq. 54.21, the utility V_{car} contains all the attributes of the car alternative. The utility V_{PT} , as we stated before, have two components:

$$V_{PT} = \beta_{EMU}W_{EMU} + \beta_{PT}x_{PT} \quad (54.22)$$

Here W_{EMU} is the expected maximum utility (EMU) of the alternatives in the nest of PT, $W_{EMU} = \ln[\exp(W_{bus}) + \exp(W_{metro})]$, x_{PT} is the vector of attributes that are taken out to estimate the binary logit model at the lower level binary model in the nest, and β_{EMU} and β_{PT} are parameters to be estimated.

54.6.2 Mixed Logit Model

Different from previous models where the parameters β in the utility function are all constant values (though the utility function has a random term ϵ), the mixed logit model allows parameters β become random variables obeying a certain distribution:

$$U_{ij} = \sum_k \beta_{ik}x_k + \epsilon_j \quad (54.23)$$

$$\beta_k \sim G(\beta|\theta)$$

Here, $G(\cdot)$ represents the density function of a certain type of distribution, with θ as a set of parameters that could describe the distribution. With this specification, the model aims to model the heterogeneity due to personal preferences. Because that each β in the utility model is a distribution rather than a constant value, the calculations of the utility of each alternatives and the probability become integral processes. That means that the choice probability of alternative j ($j = 1, 2, \dots, J$) being chosen is

$$P(y_j = 1) = \int \frac{\exp(V_j(\beta))}{\sum_{k=1}^J \exp(V_k(\beta))} G(\beta|\theta) d\beta \quad (54.24)$$

It could be noticed that the standard logit model is a special case of the mixed logit model: if the distribution $G(\beta|\theta)$ is only defined at a fixed value a , i.e., $G(\beta|\theta) = 1$ when $\beta = a$ and $G(\beta|\theta) = 0$ when $\beta \neq a$, the choice probability in Eq. 54.24 then becomes the one used in the logit model. Actually, it has been proved [30] that a mixed logit model can approximate to any degree of accuracy any true random utility model of discrete choice, given an appropriate specification of the variables and the distribution of coefficients.

The distribution $G(\beta|\theta)$ can also be discrete. Assume that β could take M distinct values, b_1, b_2, \dots, b_M . The probability for β to be value b_m , $m \in (1, 2, \dots, M)$ is $P(\beta = b_m) = p_m$. In this case, the choice probability for individual i to choose alternative j is

$$P(y_j = 1) = \sum_{m=1}^M p_m \frac{\exp(V_j(b_m))}{\sum_{k=1}^J \exp(V_k(b_m))} \quad (54.25)$$

One possible challenge for researchers is that there is a need to specify a certain type of distribution for each β if he/she plan to use a mixed logit model. Normal distribution is widely used in applications. The lognormal distribution is proved to be useful when a parameter is known to have the same sign for every individual according to domain knowledge. Uniform and triangle distributions have the advantage of being bounded on both sides, thus avoiding having too large parameters and causing unexpected problems. A more comprehensive summary could be found in [39].

Table 54.2 The first row of the original dataset

ID	FSG.1	FSO.1	TT.1	FSG.2	FSO.2	TT.2	FSG.3	FSO.3	TT.3	FSG.4	FSO.4	TT.4	FSG.5	FSO.5	TT.5	CHOICE
1007	1469	3123	594	60	26	0	309	2546	0	959	5877	0	1991	10068	0	1

54.7 Numerical Study

In this section, we use an example data to show how the three models, multinomial logit model, mixed logit model, and RRM, could be used. We implement the models in R 3.6.2.

54.7.1 General Description

The dataset we use in this section is a revealed preference (RP) dataset about shopping location choices, with 1503 shopping choices from 1074 individuals among five different alternatives. Each alternative has three attributes: “floor space groceries” (FSG), “floor space other” (FSO), and “travel time” (TT) indicating the availability of the shopping location for an individual. More details of the dataset could be found in [3].

The original dataset has 1503 rows, with each row representing a specific choice scenario and the choice made by the individual. We first convert the raw data into a new dataset whose format may be used in discrete choice models. For example, the first row of the original dataset is

In Table 54.2, attribute “ID” is the id of an individual; “FSG.1,” “FSO.1,” and “TT.1” are the three attributes of alternative 1. There are 5 alternatives, so there are 15 attributes in total. The last column “CHOICE” is the choice made by the individual.

First, we need to convert each row of the original dataset into a format as following:

We could see that the single record in Table 54.3 is converted into five rows, each representing an “true/false” binary choice for an alternative. After conversion, the original dataset with 1503 rows and 17 variables becomes a new dataset with 7515 rows and 7 attributes.

MNL

We apply a multinomial logit model (MNL) first to the dataset using the “mlogit” package in R with the following utility functions [19, 38]. Here the parameters apply to all alternatives (i.e., we do not include alternative-specific parameters for each variable):

$$U_{ji} = \beta_{FSG}FSG_{ji} + \beta_{FSO}FSO_{ji} + \beta_{TT}TT_{ji} + \epsilon_{ji} \quad (54.26)$$

Table 54.3 The converted dataset

ID	CHOICE	alt	FSG	FSO	TT	chid
1007	TRUE	1	1469	3123	594	1
1007	FALSE	2	60	26	0	1
1007	FALSE	3	309	2546	0	1
1007	FALSE	4	959	5877	0	1
1007	FALSE	5	1991	10068	0	1

Table 54.4 MNL model results

	Estimate	Std. Error	z-value	Pr(> z)
FSG	1.0595e-04	1.5040e-05	6.6889	2.248e-11***
FSO	1.1036e-05	2.2175e-06	4.9767	6.467e-07***
TT	-4.4843e-04	5.0021e-05	-8.9648	<2.2e-16***
				LL: -2305.2

*** Significant level of 0.05

The result of the multinomial logit model is shown in Table 54.4.

From Table 54.4 we could see that all the estimates are significant at 0.05 level. It is expected that the estimates for FSG and FSO are positive, and the estimate for TT is negative.

There are also various model structures for MNL. For example, we could estimate alternative-specific parameters (i.e., β s are different with different alternatives), add alternative-specific constants, or combine several alternatives together in the utility function. There are also different command that could be used other than “mlogit” in R. Readers who are interested in more details of the package may refer to its user manual.

Mixed Logit Model

With the same format of utility functions, in mixed logit model, we assume that the distribution patterns of the three attributes are all normal:

$$\begin{aligned}
 U_{ji} &= \beta_{FSG}FSG_{ji} + \beta_{FSO}FSO_{ji} + \beta_{TT}TT_{ji} + \epsilon_{ji} \\
 \beta_{FSG} &\sim N(b_{FSG}, sd_{FSG}^2) \\
 \beta_{FSO} &\sim N(b_{FSO}, sd_{FSO}^2) \\
 \beta_{TT} &\sim N(b_{TT}, sd_{TT}^2)
 \end{aligned} \quad (54.27)$$

In the results shown in Table 54.5, the estimates of the three variables are β_{FSG} , β_{FSO} , and β_{TT} in the utility functions. Meanwhile, they are also b_{FSG} , b_{FSO} , and b_{TT} that are parameters of the normal distribution. $sd.FSG$, $sd.FSO$, and $sd.TT$ are the three parameters of the standard deviations of the three distributions, respectively. As we could see, all the estimates are statistically significant at 0.05 level.

The results of the mixed logit model also include information about the random coefficients as shown in Table 54.6, including the ranges (min, mean, and max) and the quantiles (1st quantile, median, 3rd quantile) of the distributions.

RRM

In random regret minimization (RRM) model, instead of using utility functions to describe alternatives, we use regret functions:

$$R_{ji} = \sum_{k \neq j} \sum_m \ln(1 + \exp(\beta_m [x_{kmi} - x_{jmi}]))$$

$$j, k \in (1, 2, 3, 4, 5)$$

$$m \in (FSG, FSO, TT)$$
(54.28)

We use the ‘‘Apollo’’ package [40] written by Sander van Cranenburgh to implement the RRM model. We could see from Table 54.7 that except for the classical estimate, standard error, t-ratio (against 0), and p-value, there are several other columns showing the information of robust covariance estimators. Since the robust covariance estimators can tackle potential outliers in the dataset, they could be used to estimate the covariance of the real datasets or to perform outlier detection and discard/downweight some observations in the dataset. More information regarding the ‘‘Apollo’’ package and result interpretation could be seen in [24].

Table 54.5 Mixed logit model results

	Estimate	Std. Error	z-value	Pr(> z)
FSG	1.0595e-04	1.4740e-05	7.1883	6.561e-13***
FSO	1.1036e-05	2.1087e-06	5.2334	1.664e-07***
TT	-4.4843e-04	4.1971e-05	-10.6842	<2.2e-16***
sd.FSG	1.0000e-01	1.9338e-05	5171.1155	<2.2e-16***
sd.FSO	1.0000e-01	1.7770e-05	5627.3124	<2.2e-16***
sd.TT	1.0000e-01	6.5868e-05	1518.1946	<2.2e-16***
				LL: -7189.3

*** Significant level of 0.05

Table 54.6 Mixed logit model: random coefficients

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
FSG	-Inf	-0.06734302	1.059525e-04	1.059525e-04	0.06755493	Inf
FSO	-Inf	-0.06743794	1.103579e-05	1.103579e-05	0.06746001	Inf
TT	-Inf	-0.06789741	-4.484313e-04	-4.484313e-04	0.06700054	Inf

Results Comparison

Since the models do not have the same format, we do not compare the estimated β s. Instead, we summarize the log-likelihood (LL), AIC, BIC, and computational time for the three models.

From Table 54.8 we could see that the log-likelihoods (LL), AICs [1], and BICs [35] of MNL and RRM are close to each other, while RRM shows a little bit superior performance. AIC (Akaike information criterion) and BIC (Bayesian information criterion) are two estimators that deal with the trade-off between the goodness-of-fit and the simplicity of a model. Thus, they are commonly used for model selection. The formulation of AIC is

$$AIC = 2k - 2 \log(\hat{L})$$
(54.29)

where k represents the number of parameters to be estimated by the model and \hat{L} is the maximum value of the likelihood function of the model.

The formulation of BIC is a bit different from AIC:

$$BIC = \log(n)k - 2 \log(\hat{L})$$
(54.30)

where n represents the number of data points/observations/sample size.

From the comparison we could also see that mixed logit model may not be appropriate for this dataset. If we take a look at the standard deviation of the normal distribution of each variable, we could notice that the standard deviations are very large comparing with the estimates. This indicates that there is considerable heterogeneity among the individuals, or we did not choose a good distribution type to model the variability of the parameters among individuals.

54.8 Discussion

54.8.1 Limitations and Challenges

Various choice models have been proposed and applied in methodological studies and real-world applications. Though in this chapter we briefly discuss several commonly used models, there are variants of the models that are also promising, and there will be more emerging in the future to help understand, predict, and model user behavior. However, with all these models and successful implementations, there are still many questions unanswered.

Table 54.7 RRM

	Estimate	Std.err.	t.ratio	p-value	Robust std.err.	Robust t.ratio	Robust p-value
FSG	0.0680	0.0100	6.77	0.000	0.0170	3.99	0.000
FSO	0.0029	0.0011	2.79	0.005	0.0017	1.76	0.079
TT	-0.0155	0.0019	-8.35	0.000	0.0031	-4.99	0.000
							LL -2300.92

Significant level of 0.05

Table 54.8 Comparison of the three models

Model	LL	AIC	BIC	Time (s)
MNL	-2305.2	4616.494	4632.439	0
Mixed logit model	-7189.3	14390.63	14400.58	23
RRM	-2300.92	4607.841	4623.786	4

For the first, though current models with different assumptions provide various possible interpretations to choice behavior and show optimistic prediction results in existing studies, we are still not clear what circumstances are suitable for what models or which model is more accurate than other models, in a certain area of application. Moreover, to understand each model better, there might be more metrics that could be used to evaluate the performance of the models other than goodness-of-fit and prediction accuracy. We are still far away from the truth or the underlying mechanism of human choice behavior.

For the second, while most researchers have realized the importance of context dependency, there is still a lack of understanding of the influence of various types of the dependencies across space and over time. Since RRM and RAM were developed only recently (in relative sense), it is reasonable to expect more studies that will be focusing on related topics in the future.

As smart, personal devices are becoming essential items in our daily life, developments of user choice model face new challenges. A huge amount of products are bringing out individual services, in which individual preference learning and personal choice model are becoming important in implementations. However, data from an individual is often limited, both in sample size and in quality. Moreover, as the smart devices make it possible for the model to be updated in real time, the computational time of the individual choice model is required to be as short as possible to enable online applications. A more complicated learning and updating system should be designed for real-world implementation of the choice models.

Besides the models we discussed, there are also other methods that could be used to model choice behaviors, such as Hidden Markov Model, Bayesian network, or other dynamic models to estimate the state-dependent choice [6, 43, 45]. Strategies in machine learning also found useful

applications. For example, decision tree in artificial intelligence has been used to explore the underlying rules of travelers' switching decisions [37]. Recommendation systems that are frequently applied in e-commerce platforms may not be considered as choice models; the idea beneath may be inspirational especially when proposing personalized option menu for a user to choose [36].

54.8.2 Irrational Behavior

In previous models we assumed that individuals are rational or at least bounded-rational. This may not always be the case. Research in social science, psychology, and behaviors have proved that individuals have some irrational behaviors [7]. Take two famous protocols as examples:

1. Heuristic choice protocols hold that individual's choice mechanism relies on relatively little information and rule-of-thumb thinking. For example, when choosing between two alternatives, one recognized and the other not, an individual tends to choose the recognized one.
2. Social network choice protocol, which means that an individual is making choices who is not only subject to the considerations by himself/herself but also heavily impacted by other individuals who have close connection with him/her.

From these two protocols, we could see that individuals are often irrational. This provides another direction of individual choice model development, and some research have been putting efforts into it, e.g., [10]. For example, models with neural network architectures may help connect the irrational behavior with interpretable constraints and evolutionary rules [20], new assumptions to traditional normative models may provide better explanation to the mental activities when individuals are doing forced-choice tasks [31], and a new travel behavior model based on dual process may also reflect the seemingly irrational behaviors of drivers [18].

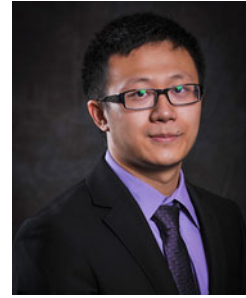
References

1. Akaike, H.: A new look at the statistical model identification. *IEEE Trans Autom Control* **19**(6), 716–723 (1974)
2. Aloulou, F.: The application of discrete choice models in transport. In: *Statistics: Growing Data Sets and Growing Demand for Statistics*, p. 85 (2018)
3. Arentze, T.A., Oppewal, H., Timmermans, H.J.: A multipurpose shopping trip model to assess retail agglomeration effects. *J. Market. Res.* **42**(1), 109–115 (2005)
4. Arrow, K.J.: Utilities, attitudes, choices: a review note. *J. Econom. Soc.* 1–23 (1958)

5. Arrow, K.J.: *Social Choice and Individual Values*, vol. 12. Yale University Press, Yale (2012)
6. Becker, F., Danaf, M., Song, X., Atasoy, B., Ben-Akiva, M.: Bayesian estimator for logit mixtures with inter- and intra-consumer heterogeneity. *Transport. Res. B: Methodol.* **117**, 1–17 (2018)
7. Becker, G.S.: Irrational behavior and economic theory. *J. Polit. Econ.* **70**(1), 1–13 (1962)
8. Belgiawan, P.F., Dubernet, I., Schmid, B., Axhausen, K.: Context-dependent models (CRRM, MuRRM, PRRM, RAM) versus a context-free model (MNL) in transportation studies: a comprehensive comparisons for swiss and german sp and rp data sets. *Transportmetrica A: Transport Sci.* **15**(2), 1487–1521 (2019)
9. Ben-Akiva, M., Bierlaire, M.: Discrete choice methods and their applications to short term travel decisions. In: *Handbook of Transportation Science*, pp. 5–33. Springer, Berlin (1999)
10. Ben-Akiva, M., McFadden, D., Train, K., Walker, J., Bhat, C., Bierlaire, M., Bolduc, D., Boersch-Supan, A., Brownstone, D., Bunch, D.S., et al.: Hybrid choice models: progress and challenges. *Market. Lett.* **13**(3), 163–175 (2002)
11. Bhat, C.R., Eluru, N., Copperman, R.B.: Flexible model structures for discrete choice analysis. In: *Handbook of Transport Modelling*, 2nd edn., pp. 75–104. Emerald Group Publishing Limited (2007)
12. Chorus, C., Arentze, T., Timmermans, H.: Random regret minimization as a choice model: theoretical and empirical comparisons with rum-modelling. In: *International Choice Modelling Conference* (2009)
13. Chorus, C.: Random regret minimization: an overview of model properties and empirical evidence. *Transport Rev.* **32**(1), 75–92 (2012)
14. Chorus, C., van Cranenburgh, S., Dekker, T.: Random regret minimization for consumer choice modeling: assessment of empirical evidence. *J. Bus. Res.* **67**(11), 2428–2436 (2014)
15. Chorus, C.G.: A generalized random regret minimization model. *Transp. Res. B: Methodol.* **68**, 224–238 (2014)
16. Chorus, C.G., Arentze, T.A., Timmermans, H.J.: A random regret-minimization model of travel choice. *Transp. Res. B: Methodol.* **42**(1), 1–18 (2008)
17. Chrzan, K., Forkner, J.: The random regret minimization choice modeling paradigm: an introduction with empirical tests. *Sawtooth Software, Research Paper Series* (2014)
18. Chung, Y.-S.: Seemingly irrational driving behavior model: the effect of habit strength and anticipated affective reactions. *Accid. Anal. Prev.* **82**, 79–89 (2015)
19. Croissant, Y., et al.: Estimation of multinomial logit models in r: the mlogit packages. R package version 0.2-2 (2012). <http://cran.r-project.org/web/packages/mlogit/vignettes/mlogit.pdf>
20. da Silva, C.F., Baldo, M.V.C.: A simple artificial life model explains irrational behavior in human decision-making. *PLoS One* **7**(5), e34371 (2012)
21. Fan, W., Yan, Z.: Factors affecting response rates of the web survey: a systematic review. *Comput. Hum. Behav.* **26**(2), 132–139 (2010)
22. Glasser, W.: *Choice Theory: A New Psychology of Personal Freedom*. HarperPerennial (1999)
23. Hausman, J.A., McFadden, D.: Specification tests for the multinomial logit model (1981)
24. Hess, S., Palma, D.: Apollo: a flexible, powerful and customisable freeware package for choice model estimation and application. *J. Choice Model.* **32**, 100170 (2019)
25. Kivetz, R., Netzer, O., Srinivasan, V.: Alternative models for capturing the compromise effect. *J. Market. Res.* **41**(3), 237–257 (2004)
26. Leong, W., Hensher, D.A.: Contrasts of relative advantage maximisation with random utility maximisation and regret minimisation. *J. Transport Econ. Policy* **49**(1), 167–186 (2015)
27. Li, D., Gao, Y., Li, R., Zhou, W.: Hybrid random regret minimization and random utility maximization in the context of schedule-based urban rail transit assignment. *J. Adv. Transp.* (2018)
28. McFadden, D.: Econometric models of probabilistic choice. *Struct. Anal. Discrete Data Econom. Appl.* 198272 (1981)
29. McFadden, D., Karlqvist, A.: Spatial interaction theory and planning models. *Model. Choice Residential Location* 75–96 (1978)
30. McFadden, D., Train, K.: Mixed mnl models for discrete response. *J. Appl. Econom.* **15**(5), 447–470 (2000)
31. McKenzie, C.R., Wixted, J.T., Noelle, D.C.: Explaining purportedly irrational behavior by modeling skepticism in task parameters: an example examining confidence in forced-choice tasks. *J. Exp. Psychol. Learn. Memory Cogn.* **30**(5), 947 (2004)
32. Parker, R.: Human behavior modeling: the necessity of narrative. In: *Computer Architecture in Industrial, Biomechanical and Biomedical Engineering*. IntechOpen (2019)
33. Popoola, O.P., Wang, K.: Video-based abnormal human behavior recognition—a review. *IEEE Trans. Syst. Man Cybern. C (Appl. Rev.)* **42**(6), 865–878 (2012)
34. Ranney, T.A.: Models of driving behavior: a review of their evolution. *Accid. Anal. Prev.* **26**(6), 733–750 (1994)
35. Schwarz, G., et al.: Estimating the dimension of a model. *Ann. Stat.* **6**(2), 461–464 (1978)
36. Song, X., Danaf, M., Atasoy, B. and Ben-Akiva, M.: Personalized menu optimization with preference updater: a Boston case study. *Transp. Res. Record* **2672**(8), 599–607 (2018)
37. Tang, L., Xiong, C., Zhang, L.: Decision tree method for modeling travel mode switching in a dynamic behavioral process. *Transp. Plan. Technol.* **38**(8), 833–850 (2015)
38. Train, K., Croissant, Y.: exercise 1: multinomial logit model. <https://cran.r-project.org/web/packages/mlogit/vignettes/e1mlogit.html>. Accessed 04 Feb 2020
39. Train, K.E.: *Discrete Choice Methods with Simulation*. Cambridge University Press, Cambridge (2009)
40. van Cranenburgh, S.: Advanced random regret minimization models. <https://www.advancedrrmmodels.com/classical-rrm>. Accessed 04 Feb 2020
41. Sander van Cranenburgh, Cristian Angelo Guevara, and Caspar G Chorus. New insights on random regret minimization models. *Transpor. Res. A: Policy Practice* **74**, 91–109 (2015)
42. Williams, H.C.W.L.: On the formation of travel demand models and economic evaluation measures of user benefit. *Environ. Plan. A* **9**(3), 285–344 (1977)
43. Xiong, C., Zhang, L.: Dynamic travel mode searching and switching analysis considering hidden model preference and behavioral decision processes. *Transportation* **44**(3), 511–532 (2017)
44. Zhang, J.: A generalized relative utility based choice model with multiple context dependencies. In: *Proceedings of the International Choice Modelling Conference*, pp. 3–5 (2013)
45. Zhu, Z., Chen, X., Xiong, C., Zhang, L.: A mixed bayesian network for two-dimensional decision modeling of departure time and mode choice. *Transportation* **45**(5), 1499–1522 (2018)



Ms. Xi Zhu received the B.S. degree in Civil Engineering from Tsinghua University, Beijing, China, in 2013 and the M.S. degree in Transportation Engineering from University of Washington, Seattle, WA USA, in 2015. She is currently pursuing the Ph.D. degree in Transportation Engineering at University of Washington, Seattle, WA, USA. Her research interests include human behavior modeling and experimental design.



Dr. Shuai Huang received the B.S. degree in statistics from the University of Science and Technology of China in 2007, and the Ph.D. degree in industrial engineering from Arizona State University, Tempe, AZ, USA, in 2012. He is currently an Associate Professor with the Department of Industrial & Systems Engineering, University of Washington, Seattle, WA, USA. His research interests include statistical learning and data mining with applications in healthcare and manufacturing.



Hainan Zhang

Contents

55.1	Introduction	1090
55.2	Weighted Threshold Voting System	1091
55.2.1	Model Assumptions.....	1091
55.2.2	Decision Rule.....	1091
55.2.3	Model Formulation.....	1091
55.2.4	Numeric Examples of System Reliability.....	1092
55.3	Reliability Analysis	1093
55.3.1	Reliability Approximation.....	1093
55.3.2	Numeric Examples of System Reliability Approximation.....	1094
55.3.3	Time-Dependent Reliability Function.....	1094
55.3.4	Optimal Stopping Time at Minimal Cost.....	1096
55.4	Weighted Threshold Indecisive Voting System	1098
55.4.1	Model Assumptions.....	1099
55.4.2	Indecisive Decision Rule.....	1099
55.4.3	Indecisive Model Formulation.....	1099
55.4.4	Numerical Example of Indecisive System Reliability.....	1100
55.4.5	Sensitivity Analysis on WVS.....	1102
55.5	Other Related Works	1105
55.5.1	Model-Related Studies.....	1105
55.5.2	Application-Related Studies.....	1106
55.6	Conclusion and Future Research	1107
	References	1107

Abstract

Voting is an ancient method for a group such as a meeting or an electorate to make a collective decision or express an opinion after some discussions, deliberations, or election campaigns. Participants who give an option or choose a candidate are called voters; therefore, a simplest voting system consists certain number of qualified voters and

candidates. The most common application in weighted voting systems is, for example, the US Electoral College, where the number of electoral votes for each state is based upon its population. This chapter states the modeling of threshold weighted voting systems and the dynamic analysis of two terms including indecisive effect and known/unknown inputs. The system reliability models presented in the chapter are based on the assumptions that the system operates as two types of input values (0, 1) and three types of output values (0, 1, x) with three types of errors, and the components are unequally weighted and subject to three failure-modes (stuck-at-0, stuck-at-1, stuck-at- x). For any weighted voting system, a decision rule is required although different rules may result different system performance in terms of the system reliability. For instance, the current decision rule of US Electoral College is who wins the election obtaining at least 270 electoral votes, which results four former US presidents won the elections with less national populate votes than their opponents in history.

In general, the weighted voting system (WVS) consists of n units assigned with individual weights, each of which provides a binary decision (0 or 1) or abstains (x) from voting. A generic decision rule can be defined as the system output is 1 if the cumulative weight of all 1-opting units is at least a prespecified threshold τ of the cumulative weight of all nonabstaining units. If the indecisive effect is considered, weights of abstaining units can be added in the decision rule such as the system output is 1 if the cumulative weight of all 1-opting units is at least a prespecified threshold τ of the sum of all nonabstaining units and prespecified indecisive parameter θ of all abstaining units. The system fails if the generated output is not equal to its original input. Recent research results indicate that, under specified assumptions, multiple approaches can be used to quantify the reliability of the weighted voting system. This chapter demonstrates the development of decision rules and the evolution of approaches of generating reliability function. Some related works are addressed to provide a

H. Zhang (✉)
 Department of Industrial & System Engineering, Rutgers University –
 New Brunswick, Piscataway, NJ, USA
 e-mail: h275@rutgers.edu; h275@scarletmail.rutgers.edu

full picture of WVS, and some future works are proposed to attract attentions.

Keywords

Reliability · Decision rule · Weighted voting · Decision-making · Modeling

55.1 Introduction

Weighted voting system (WVS) is one of the modern tools used in decision-making by assigning different powers to voters. A weighted voting system is characterized by three basic parts – *voters*, *weights*, and the *voting rule*. The voters can be considered as a group of n individuals, systems, or components (v_1, v_2, \dots, v_n) that each one can generate an independent decision. Each voter's weight w_i is the power of its individual decision counts. The rule is the minimum requirement of votes to make the final decision. Weighted voting is a comprehensive way of decision-making, and it has many significant applications in a wide range of systems including communication systems [1], financial systems [2], election systems [3], and medical systems [4]. Thus, more specific aspects and assumptions are naturally generated, which will be discussed briefly in the end of this chapter.

Two important aspects of *voters* include single voter's reliability and failure modes. Each voter plays an independent role in providing its own output, and its reliability is a value to quantify the probability of making correct output. Therewith, the fact of imperfect voter results in failure rate; however, different failure modes may result from wrong or absent output and input patterns. System design is then distinguished; therefore, system optimization is affected. *Levitin* [5] analyzed the optimization of systems consisting of voting units with limited availability and presented two ways to improve system reliability, where a method of taking information about units' availability into procedure for system optimization was approached. *Pham* and *Malon* [6] studied reliability optimal design of systems with competing failure modes. They achieved optimal parameterizations to provide maximum system reliability and minimum system cost. *Pham* [7] then studied reliability for dynamic configurations of systems with three failure modes. He presented the comparisons of reliability modeling function and optimal parameterization from two failure modes to three failure modes.

Unequal *weights* in multiunit systems are more frequently considered in many cases. The weight of each voter represents its decision counts in the whole system, especially assigning strategy is more careful when indecisive effect is considered. In classic models, weights of voters with no decision do not count; however, in modern models, indecisive

weights are considered and become obstacles of decision-making. Those obstacles can either be pros or cons, and there are typical adjustments in favor of goals from different cases. In 1976, *Lucas* [8] studied the relations between voting powers and weights, and he pointed out that not all inequalities could be canceled out by weighted voting and power was not trivial from one's strength as measured by the number of votes; therefore, his results motivated independent research on weights. *Levitin* [9] designed an asymmetric WVS in which each voting unit had two weights, where the weight was applied if the unit's decision was 0 and the second weight was applied if the unit's decision was 1. The idea of the asymmetry of unit weights allows the designer to take advantage of the knowledge of statistical asymmetry of voting units. *Aziz*, *Paterson*, and *Leech* [10] designed a WVS with a desirable distribution of power, where they presented an algorithm to compute an integer weight vector for target Banzhaf power indices.

The *Voting rule* is the fundamental law for a WVS to make the final decision. In classical models, electoral college, the voting rule is nothing but a threshold value. In modern models, voting rule consists of weights and parameters, human organizations; decision criterion is usually more complicated as many factors are involved in the final decision. *Nordmann* and *Pham* [11] built a dynamic system by considering weights along with threshold in the decision rule, where applicable assumptions were raised in case studies. *Levitin* [12] focused on threshold optimization; in such a procedure, he studied the method of finding threshold values to minimize the system cost. In 2014, *Posner* and *Sykes* [13] showed that all major voting systems were controversial in different ways which was resulted from the difficulty of designing a voting system that both allowed efficient decisions and protected the legitimate interests of voters. And they provided a new type of voting system, quadratic voting, which could resolve these problems theoretically but was unlikely to be implemented any time soon.

This chapter presents recent results on modeling reliability of voting systems consisting of n unequally weighted components with three failure modes in human organizations. The early work is initially studied by *Nordmann* and *Pham* in [11, 14] where the fundamental framework was constructed, and application gaps were also raised. Later, the modern research studied by *Xie* and *Pham* [15] improved methodology that a general recursive reliability function was presented. Recently, *Zhang* and *Pham* [16] have presented a weighted indecisive-voting system considering indecisive effect. The results illustrate ignored insights into system behavior in terms of parameterizations and reliability sensitivity in terms of input patterns. This chapter summaries these results and points out the motivation of some extended research on controlled voting when lacking prior information of inputs.

55.2 Weighted Threshold Voting System

Consider weighted voting systems model studied in [15]. The model descriptions and assumptions are given as follows. Specifically, we use human organization (HO) systems as a general illustrative example to represent the applied techniques of the generalized WVS.

55.2.1 Model Assumptions

- WVS is a decision-making system where a final decision is made based on any **given** input, *e. g.*, proposal in HO system, denoted by 1 (good) or 0 (bad).
- There are n voters in the WVS.
- All voters are statistically independent and nonidentical.
- Each voter makes a subdecision, and the final decision 1 (accept), 0 (reject), or x (no decision) is based on n subdecisions.
- For any voter v_i , an independent weight w_i is assigned.
- For any voter v_i , its probability of making correct subdecision is $p_{i,00}$ (reject bad proposal) and $p_{i,11}$ (accept good proposal).
- Each voter has probability of making wrong decision subject to three types of errors.
 1. Type 1: Accept the one should be rejected, $p_{i,01}$.
 2. Type 2: Reject the one should be accepted, $p_{i,10}$.
 3. Type 3: No decision on the one should be rejected, $p_{i,0x}$; no decision on the one should be accepted, $p_{i,1x}$.
- A decision rule is designed with a predefined value of threshold, τ , incorporating all subdecisions to generate the final decision.

55.2.2 Decision Rule

To illustrate how such systems fit into the framework of general models, Xie and Pham [15] described the problem in an HO system. Assume the proposal is the input of either acceptable and unacceptable and the information is implicitly contained in the material that each voter is going to review. Thus, via the disguised form, each voter gets an input either 1 (acceptable) or 0 (unacceptable). All voters are provided with the same input that depends on the actual proposal where a decision is to be made, and therefore produces a subdecision that is 1 (accept), 0 (reject), or x (indecisive). Ideally, the output from each voter should equal to its input; however, under the fact of imperfect voters, each voter is subject to errors described above.

Consider a weighted threshold voting system of n voters with the weight w_i for voter i , where I_1 is the index set of all voters with favorable outcomes for the system input, I_x is

the index set of all voters without any individual decision, and $\tau \in (0, 1]$ is the threshold value. The system collects all subdecisions from all voters and generates a joint system output based on the following decision rule [11, 14, 17]. The system output is “1” (acceptable) if and only if,

$$\sum_{i \in I_1} w_i \geq \tau \sum_{i \notin I_x} w_i \quad (55.1)$$

and it is “0” otherwise.

This model is a dynamic threshold weighted voting system subject to two failure modes as it does not consider weights for voters with no decision. Failure mode 1 means that the system accepts an input ought to be rejected, and failure mode 2 means that the system rejects an input ought to be accepted. Thus, the reliability of the system, R_{sys} , is that probability of the system making a correct decision for any given input.

$$R_{\text{sys}} = \Pr \{ \text{the system makes the correct decision} \}$$

Nordmann and Pham [11, 14] proposed this decision rule by developing a mathematical model with two restrictions on the scale of weights and the value of threshold, which resulted from the immense combinatorial complexity and thus cannot be applied directly to many real systems due to the complexity of computations. Simply, two restrictions on the generality of parameters in their model are the key obstacles from practical usage. Therefore, to formulate the model, an advanced methodology is induced. Later, Levitin [17] used this decision rule to estimate WVS reliability with universal generating function technique. Although his algorithm improved the calculation procedure, its functionality was still limited due to the intrinsic shortcomings of the algorithm until Xie and Pham [15].

55.2.3 Model Formulation

Xie and Pham [15] applied a simple recursive formula to calculate the reliability of the weighted voting systems without restrictions on both weights and threshold values. Define a random variable Z_i where $i = 1, 2, \dots, n$ as

$$Z_i = \begin{cases} 1 - \tau & \text{if } i \in I_1 \\ 0 & \text{if } i \in I_x \\ -\tau & \text{if } i \in I_0 \end{cases} \quad (55.2)$$

By rewriting Eq. (55.1), the system output is “1” if and only if,

$$\sum_{i \in I_1} w_i - \tau \sum_{i \notin I_x} w_i \geq 0 \quad (55.3)$$

To simplify Eq. (55.3) for clear interpretation, replace (55.2) into the second sum; it can be disaggregated and rearranged as follows,

$$\sum_{i \in I_1} (1 - \tau) w_i - \sum_{i \in I_0} \tau w_i \geq 0 \iff \sum_{i=1}^n w_i Z_i \geq 0 \tag{55.4}$$

Thus, each voter i contributes a positive value $(1 - \tau)w_i$ if it votes acceptance, and a negative value $-\tau w_i$ if it votes rejection toward to the final WVS score, where I_1 , I_0 , and I_x represent the index sets of voters with the corresponding decisions. Further, they define two probability functions.

$$Q_n(s) = \Pr\left(\sum_{i=1}^n w_i Z_i \geq s \mid P = 1\right)$$

$$\tilde{Q}_n(s) = \Pr\left(\sum_{i=1}^n w_i Z_i < s \mid P = 0\right)$$

Then, the probability of the system output $S = 1$ given the input $P = 1$ is,

$$Q_n(0) = \Pr\left(\sum_{i=1}^n w_i Z_i \geq 0 \mid P = 1\right) = \Pr(S = 1 \mid P = 1)$$

And the probability of the system output $S = 0$ given the input $P = 0$ is,

$$\tilde{Q}_n(0) = \Pr\left(\sum_{i=1}^n w_i Z_i < 0 \mid P = 0\right) = \Pr(S = 0 \mid P = 0)$$

Therefore, the reliability of such a weighted threshold voting system with n voters is,

$$R_{\text{sys}} = Q_n(0) \Pr(P = 1) + \tilde{Q}_n(0) \Pr(P = 0) \tag{55.5}$$

where $Q_n(s)$ and $\tilde{Q}_n(s)$ can be written in terms of recursive functions as follows:

For $n \geq 1$,

$$Q_1(s) = \begin{cases} 1 & \text{if } s \leq -\tau w_1 \\ p_{1,11} + p_{1,1x} & \text{if } -\tau w_1 < s \leq 0 \\ p_{1,11} & \text{if } 0 < s \leq (1 - \tau) w_1 \\ 0 & \text{if } s > (1 - \tau) w_1 \end{cases}$$

$$\tilde{Q}_1(s) = \begin{cases} 0 & \text{if } s \leq -\tau w_1 \\ p_{1,00} & \text{if } -\tau w_1 < s \leq 0 \\ p_{1,00} + p_{1,0x} & \text{if } 0 < s \leq (1 - \tau) w_1 \\ 1 & \text{if } s > (1 - \tau) w_1 \end{cases}$$

For $n \geq 2$,

$$Q_n(s) = Q_{n-1}(s_n^-) p_{n,11} + Q_{n-1}(s) p_{n,1x} + Q_{n-1}(s_n^+) p_{n,10}$$

$$\tilde{Q}_n(s) = \tilde{Q}_{n-1}(s_n^-) p_{n,11} + \tilde{Q}_{n-1}(s) p_{n,1x} + \tilde{Q}_{n-1}(s_n^+) p_{n,10}$$

where $s_n^- = s - (1 - \tau) w_n$ and $s_n^+ = s + \tau w_n$. Given $\Pr(P = 1) + \Pr(P = 0) = 1$, Eq. (55.5) indicates that the reliability value R_{sys} has the range $[R_{\text{sys}}^{\min}, R_{\text{sys}}^{\max}]$, where $R_{\text{sys}}^{\min} = \min\{Q_n(0), \tilde{Q}_n(0)\}$ and $R_{\text{sys}}^{\max} = \max\{Q_n(0), \tilde{Q}_n(0)\}$. The following numeric examples are illustrated to show how the system works.

55.2.4 Numeric Examples of System Reliability

Example 55.1 Suppose $\Pr(P = 1) = \Pr(P = 0) = 0.5$. Consider the following three extreme cases:

- (a) The output of the system is 1.
- (b) The output of the system is 0.
- (c) The output of the system is either 0 or 1 with equal probability.

For simplicity and without loss of generality, it is assumed that there is only one voter in the system, $n = 1$. In this case, the probability that the system output of 0 or 1 is $0.5(p_{1,00} + p_{1,10})$ or $0.5(p_{1,11} + p_{1,01})$, respectively. And therefore, the system reliability is $R_{\text{sys}} = 0.5(p_{1,00} + p_{1,1x} + p_{1,11})$.

In case (a), the system output is 1, which requires $0.5(p_{1,11} + p_{1,01}) = 1$. It implies that $p_{1,11} = p_{1,01} = 1$ and other $p_{1,vv'}$ should be 0, which leads the system reliability of case (a) is $R_{\text{sys}} = 0.5$. And the similar situation can be applied on case (b); thus, the system reliability of case (b) is also $R_{\text{sys}} = 0.5$. Note that the reliability of a system is defined as the probability of making a correct decision. If the input is either 0 or 1 with probability 0.5 of each, then the system will have an equal chance to make the right decision in either case (a) or (b), although the probability for case (a) or (b) happens at different time.

In case (c), the output of the system is either 0 or 1 with equal probability, so it implies that $0.5(p_{1,00} + p_{1,10}) = 0.5(p_{1,11} + p_{1,01}) = 0.5$. Thus, $p_{1,00} = 1 - p_{1,01} = p_{1,11} = 1 - p_{1,10}$ and $p_{1,0x} = p_{1,1x} = 0$, and the system reliability is $R_{\text{sys}} = p_{1,00} = 1 - p_{1,01} = p_{1,11} = 1 - p_{1,10}$, which may not necessary be 0.5. And $R_{\text{sys}} = 0.5$ only if $p_{1,00} = p_{1,01} = p_{1,11} = p_{1,10} = 0.5$.

Example 55.2 Suppose we have $n = 4$ voters in the HO system described above and the probability matrix of their judgement errors along with weights is distributed as follows.

$$p_{1,10} = 0.17, p_{1,1x} = 0.14, p_{1,01} = 0.15, p_{1,0x} = 0.10.$$

$$p_{2,10} = 0.14, p_{2,1x} = 0.10, p_{2,01} = 0.11, p_{2,0x} = 0.14$$

$$p_{3,10} = 0.11, p_{3,1x} = 0.12, p_{3,01} = 0.16, p_{3,0x} = 0.12$$

$$p_{4,10} = 0.15, p_{4,1x} = 0.12, p_{4,01} = 0.10, p_{4,0x} = 0.15$$

$$w_1 = 2.0, w_2 = 1.0, w_3 = 1.5, w_4 = 1.0$$

Given a low threshold value $\tau = 0.2$,

$$R_{\text{sys}}(\tau = 0.2) = \begin{cases} 0.9915 & \text{if } \Pr(P = 1) = 1 \\ 0.6567 & \text{if } \Pr(P = 0) = 1 \end{cases}$$

Given a medium threshold value $\tau = 0.5$,

$$R_{\text{sys}}(\tau = 0.5) = \begin{cases} 0.9214 & \text{if } \Pr(P = 1) = 1 \\ 0.9178 & \text{if } \Pr(P = 0) = 1 \end{cases}$$

Given a high threshold value $\tau = 0.8$,

$$R_{\text{sys}}(\tau = 0.8) = \begin{cases} 0.6549 & \text{if } \Pr(P = 1) = 1 \\ 0.9918 & \text{if } \Pr(P = 0) = 1 \end{cases}$$

To quantify the system performance, reliability is calculated under different cases of τ . System has inverse reacts under two extreme values of threshold when $\tau = \{0.2, 0.8\}$ under different types of inputs. However, the medium value of threshold $\tau = 0.5$ results a reliable system with equal performance under both types of inputs.

55.3 Reliability Analysis

55.3.1 Reliability Approximation

For practical reason, Xie and Pham [15] studied an approach of reliability approximation although C code is programed to calculate the reliability automatically if all necessary values are initialized, which aims to provide a quick check of the system reliability if the exact computation is not required.

The possible combinations of votes in a weighted voting system are defined in Eq. (55.1) on which the size of the sample space of the random variable $Y_n = \sum_{i=1}^n w_i Z_i$ is 3^n . Thus, without any further assumptions on both w_i and q_i , any algorithm for computing the system reliability involves up to 3^n terms, which will result computation difficulties. To solve this problem, they proposed an accurate large-sample approximation formula for evaluating the system reliability R_{sys} . The formula is developed based on the saddle point

approximation technique by Barnforff-Nielsen and Cox [18], and it works well for small size of samples. The computations of the proposed approach are straightforward for simple linear equations while techniques are needed for nonlinear equations when locating the saddle points. For any size n , the approximation values of R_{sys} can be calculated instantly in route.

Define a similar notation $Z_i^* = Z_i + \tau$, where Z_i is defined in Eq. (55.2). Given the input $P = 1$, the logarithm of the conditional moment generating function, $\sum_{i=1}^n w_i Z_i^*$, is defined as

$$K_n(u) = \sum_{i=1}^n \log(p_{1,11} e^{uw_i} + p_{1,1x} e^{u\tau w_i} + p_{1,10})$$

Then, the solution point $u = \hat{u}_n$, which solves the first derivative of $K_n'(u) = \tau \sum_{i=1}^n w_i$, is called a saddle point. Similarly, given the system input $P = 0$, the conditional moment generating function of $\sum_{i=1}^n w_i Z_i^*$ is denoted as,

$$\tilde{K}_n(u) = \sum_{i=1}^n \log(p_{1,01} e^{uw_i} + p_{1,0x} e^{u\tau w_i} + p_{1,00})$$

By solving the first derivative equation $\tilde{K}_n'(u) = \tau \sum_{i=1}^n w_i$, the saddle point \tilde{u}_n is located.

Now let $\hat{e}_n = \hat{u}_n * [K_n''(\hat{u}_n)]^{0.5}$ and $\hat{f}_n = \text{sign}(\hat{u}_n) * |\sqrt{2}[\hat{u}_n K_n'(\hat{u}_n) - K_n(\hat{u}_n)]^{0.5}|$, and $\tilde{e}_n = \tilde{u}_n * [K_n''(\tilde{u}_n)]^{0.5}$ and $\tilde{f}_n = \text{sign}(\tilde{u}_n) * |\sqrt{2}[\tilde{u}_n \tilde{K}_n'(\tilde{u}_n) - \tilde{K}_n(\tilde{u}_n)]^{0.5}|$. Suppose existing small positive constants $\varepsilon, \delta_1, \delta_2 > 0$ and a large positive number $M < \infty$, such that $\varepsilon < w_i < M$ and $\delta_1 < 1 - p_{1,vv'} < 1 - \delta_2$ for any $v \neq v'$. Therefore, the approximation results for $Q_n(0)$ and $\tilde{Q}_n(0)$ can be represented as follows.

$$Q_n(0) = \Pr\left(\sum_{i=1}^n w_i Z_i \geq 0 \mid P = 1\right) = 1 - \psi(\hat{e}_n) + \phi(\hat{e}_n) \left[\frac{1}{\hat{e}_n} - \frac{1}{\hat{f}_n} + O(n^{-1.5}) \right]$$

$$\tilde{Q}_n(0) = \Pr\left(\sum_{i=1}^n w_i Z_i < 0 \mid P = 0\right) = \psi(\hat{e}_n) - \phi(\hat{e}_n) \left[\frac{1}{\hat{e}_n} - \frac{1}{\hat{f}_n} + O(n^{-1.5}) \right]$$

where ψ and ϕ are the cumulative distribution function and the density function of a standard normal random variable, respectively, and the notation big $O(n^{-1.5})$ represents the remaining terms in the parenthesis which tend to 0 at a certain speed equal or faster than $n^{-1.5}$ as $n \rightarrow \infty$.

The error terms are at the rate of $O(n^{-1.5})$, and the R_{sys} is at most at the order of $O(n^{-1.5})$. When n is large, these approximations are extremely accurate.

55.3.2 Numeric Examples of System Reliability Approximation

Example 55.3 Table 55.1 illustrates the reliabilities calculated for a weighted voting system with $n = 5, 10, 15, 20, 25, 40, 100,$ and 300 voters. The parameters of the error probabilities are randomly resampled from the base set of 4 voters in Example 55.2. The second column is the calculated values of $Q_n(0)$ using the recursive formulas in subsection of Model Formulation, and the third column lists the approximation values of $Q_n(0)$ calculated using the recursive formulas in subsection of Reliability Approximation. The values in the fourth column are the calculated differences between the values in the second and third columns. The fifth, sixth, and seventh columns show the corresponding values related to $\tilde{Q}_n(0)$, respectively. And the last three columns are values related to the system reliability R_{sys} given $\Pr(P = 1) = 0.7$, where more favorable inputs are chosen in the illustrated examples.

C code takes less than a minute to finish each calculation up to $n = 20$. When $n = 25$, it takes about 3 hours to finish calculating the exact values of R_{sys} ; thus, they did not try it when $n > 25$. The accuracy increases as n grows bigger as expected, especially when $n \geq 20$.

From the calculations of differences between exact and approximated values, one can see that the approximation approach is a relatively good way to identify the system performance even if when the size is small. As expected, as n approaches to a big value, the approximation of the system reliability has no difference from the exact value. And this methodology is strongly recommended under certain constraints of budget controls or time-consuming issues in some policy-related systems.

55.3.3 Time-Dependent Reliability Function

To this end, no time limitation is set up for reliability evaluation; however, to make the model more applicable in practice, the requirement of a limited time for decision-making is appropriate. Along with voters' expertise, limited time will

affect the results of their decisions. Xie and Pham [15] assumed time-dependent functions on reliability and error rate for each voter; $p_{i,vv'}(t)$ is a function of time t , where t is the time up to a decision for any voter. That is, voters must make the decision before time t and the WVS will produce an output based on subdecisions from all voters. And this design can be extended to more general cases with some additional clarifications on model assumptions.

They considered a typical pattern which is proper in real cases, where the error probabilities $p_{i,vv'}(t)$ for any $v \neq v'$ are decreasing functions of t . That is, any voter is less reliable at the beginning as time is limited; however, the probability of all voters making wrong decisions would decrease after a certain time. If sufficient time is allowed, the probability of wrong decisions would reduce to a minimal level. Figure 55.1 illustrates an example of time-dependent functions for four different failure rates. The error probability decreases slowly from the maximum value and reaches to the minimum value after a certain period. The curves in Fig. 55.1 are from the following formula. For $t > 0$ where $v \neq v', v = 0, 1,$ and $v' = 0, 1, x$.

$$p_{i,vv'}(t) = p_{i,vv'}^{\min} + (1 + e^{-b_{i,vv'}}) \frac{(p_{i,vv'}^{\max} - p_{i,vv'}^{\min}) e^{-a_{i,vv'}t + b_{i,vv'}}}{1 + e^{-a_{i,vv'}t + b_{i,vv'}}} \quad (55.6)$$

where $a_{i,vv'} > 0$ decides the pace of $p_{i,vv'}(t)$ reduces as t increases, and $b_{i,vv'} > 0$ decides the location of the curve, and two constants $0 < p_{i,vv'}^{\min} < p_{i,vv'}^{\max} < 1$ for $v \neq v'$ are lower and upper bounds for $p_{i,vv'}(t)$ for the entire domain. When $p_{i,vv'}^{\min} = 0$ and $p_{i,vv'}^{\max} = e^{b_{i,vv'}} / (1 + e^{b_{i,vv'}})$, formula (55.6) becomes to the standard logistic function for sigmoid curves so that value of $p_{i,vv'}$ is between lower and upper bounds for all $t > 0$, and $p_{i,vv'}(0) = p_{i,vv'}^{\max}$.

The reliability of the threshold voting systems can be quantified under the time-dependent situation by defining $Q_{n,t}(s)$ and $\tilde{Q}_{n,t}(s)$ similarly as $Q_n(s)$ and $\tilde{Q}_n(s)$. Under such assumptions, $Q_{n,t}(0)$ and $\tilde{Q}_{n,t}(0)$ are nondecreasing which leads the time-dependent system reliability $R_{sys}(t)$ be nondecreasing of t . An example gives an intuitive vision of how it works on WVS.

Table 55.1 Comparison of exact and approximate system reliability

n	$Q_n(0)$			$\tilde{Q}_n(0)$			R_{sys}		
	Exact	Approx	Diff	Exact	Approx	Diff	Exact	Approx	Diff
5	0.9566	0.9733	-0.0167	0.9686	0.9497	0.0189	0.9602	0.9662	-0.0060
10	0.9919	0.9953	-0.0034	0.9966	0.9944	0.0022	0.9930	0.9945	-0.0015
15	0.9983	0.9990	-0.0007	0.9993	0.9989	0.0004	0.9986	0.9990	-0.0004
20	0.9997	0.9998	-0.0001	0.9999	0.9998	0.0001	0.9997	0.9998	-0.0001
25	0.9999	0.9999	0	0.9999	0.9999	0	0.9999	0.9999	0
40	-	0.9999	-	-	0.9999	-	-	0.9999	-
100	-	0.9999	-	-	1	-	-	0.9999	-
300	-	1	-	-	1	-	-	1	-

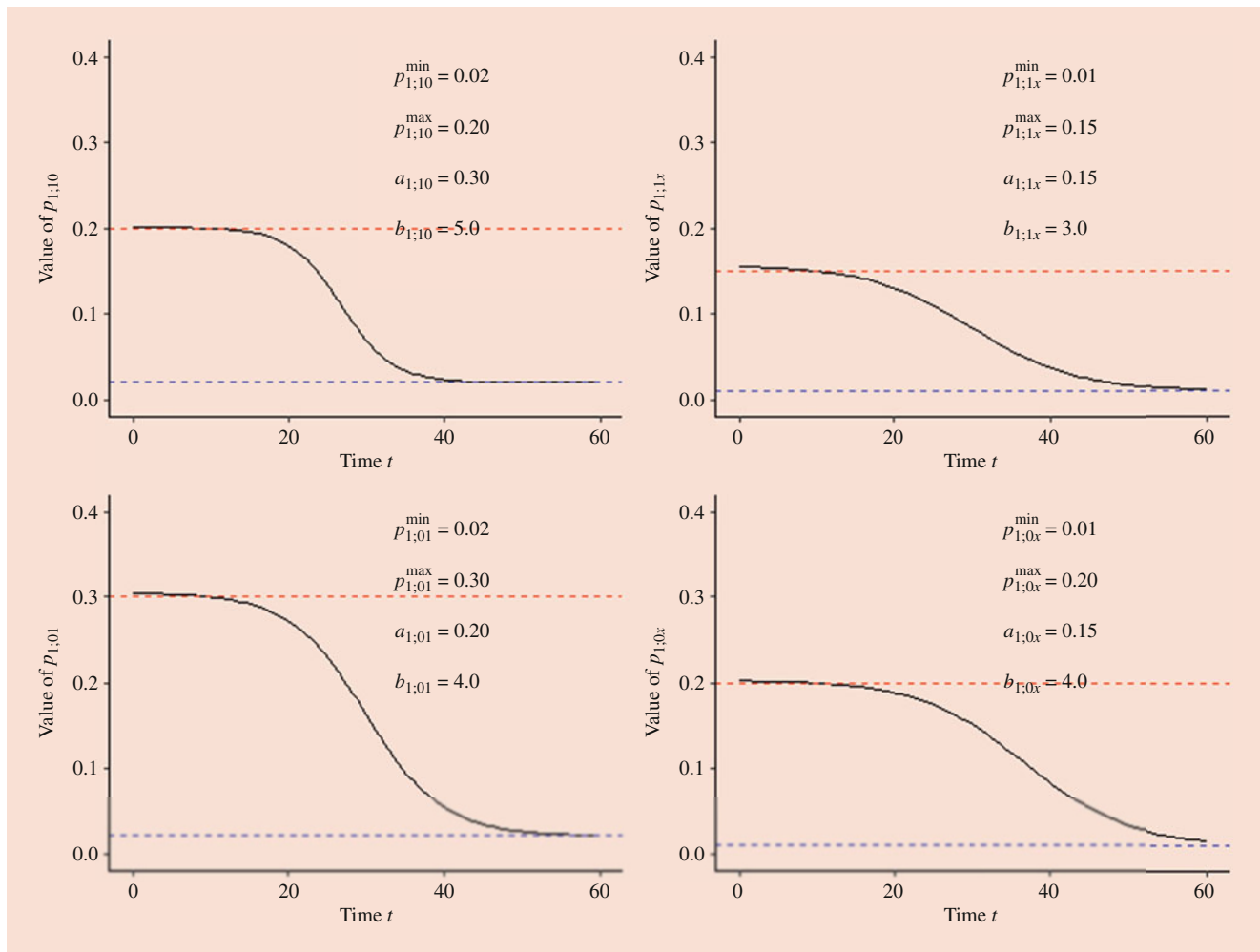


Fig. 55.1 Time-dependent functions of four failure rates

Example 55.4 Suppose a system has $n = 3$ voters. Assume their failure rates as functions of time t can be modeled in the form of (55.6), and the parameters are as follows,

Voter 1 : $(p_{1,10}^{\min}, p_{1,10}^{\max}) = (0.02, 0.20)$, $a_{1,10} = 0.30$, $b_{1,10} = 5.0$
 $(p_{1,1x}^{\min}, p_{1,1x}^{\max}) = (0.01, 0.15)$, $a_{1,1x} = 0.15$, $b_{1,1x} = 3.0$
 $(p_{1,01}^{\min}, p_{1,01}^{\max}) = (0.02, 0.30)$, $a_{1,01} = 0.20$, $b_{1,01} = 4.0$
 $(p_{1,0x}^{\min}, p_{1,0x}^{\max}) = (0.01, 0.20)$, $a_{1,0x} = 0.15$, $b_{1,0x} = 4.0$
 $w_1 = 2.0$

Voter 2 : $(p_{2,10}^{\min}, p_{2,10}^{\max}) = (0.00, 0.20)$, $a_{2,10} = 0.30$, $b_{2,10} = 4.5$
 $(p_{2,1x}^{\min}, p_{2,1x}^{\max}) = (0.01, 0.15)$, $a_{2,1x} = 0.15$, $b_{2,1x} = 2.3$
 $(p_{2,01}^{\min}, p_{2,01}^{\max}) = (0.01, 0.23)$, $a_{2,01} = 0.20$, $b_{2,01} = 4.5$
 $(p_{2,0x}^{\min}, p_{2,0x}^{\max}) = (0.00, 0.18)$, $a_{2,0x} = 0.15$, $b_{2,0x} = 5.0$
 $w_2 = 1.0$

Voter 3 : $(p_{3,10}^{\min}, p_{3,10}^{\max}) = (0.01, 0.20)$, $a_{3,10} = 0.25$, $b_{3,10} = 4.0$
 $(p_{3,1x}^{\min}, p_{3,1x}^{\max}) = (0.01, 0.10)$, $a_{3,1x} = 0.20$, $b_{3,1x} = 2.0$
 $(p_{3,01}^{\min}, p_{3,01}^{\max}) = (0.00, 0.10)$, $a_{3,01} = 0.15$, $b_{3,01} = 4.5$
 $(p_{3,0x}^{\min}, p_{3,0x}^{\max}) = (0.01, 0.20)$, $a_{3,0x} = 0.20$, $b_{3,0x} = 4.0$
 $w_3 = 1.5$

For three different options of threshold value $\tau = 0.3, 0.5$, and 0.7 , the time-dependent reliabilities $R_{\text{sys}}(t)$ at time t are calculated. Figure 55.2 shows six time-dependent reliability curves against time t . For simplicity, only two types of curves are provided. One curve is the reliability of input $P = 0$, and the other one is of input $P = 1$, respectively.

- In the first subplot with $\tau = 0.3$, the system reliability R_{sys} when $P = 1$ is always larger than $P = 0$, and its maximum value when $P = 1$ is clearly larger than $P = 0$.

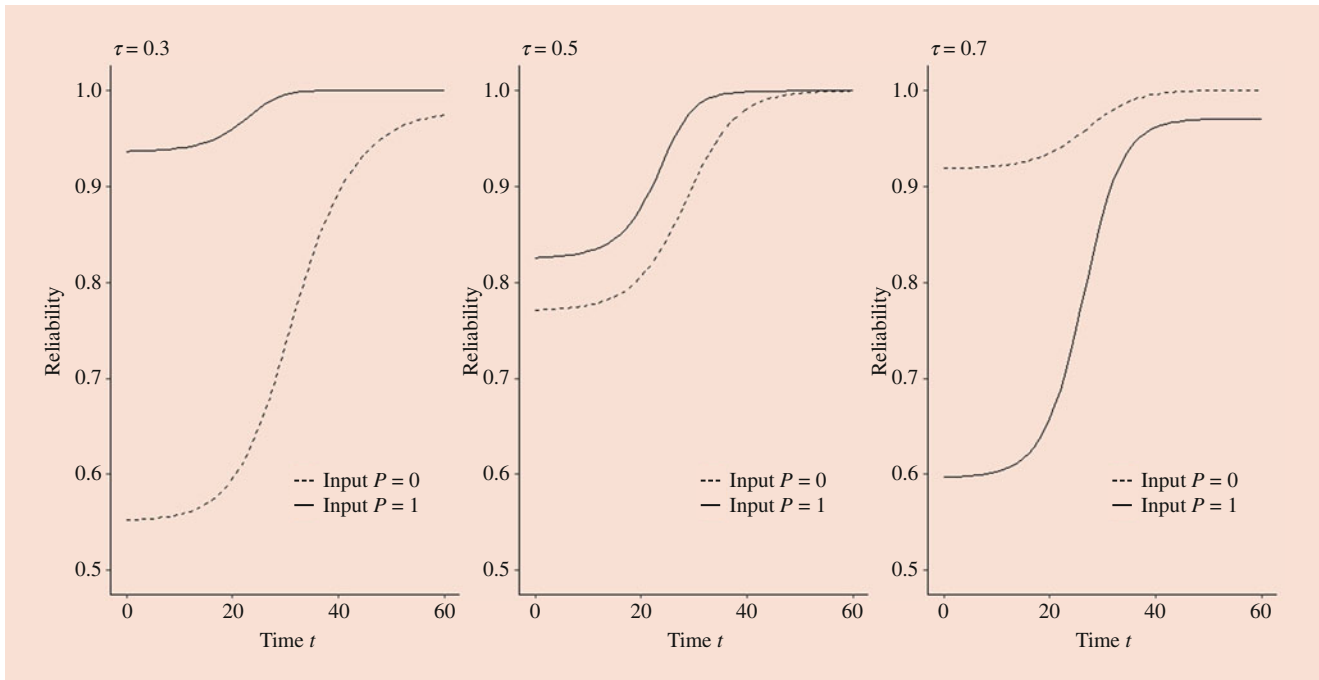


Fig. 55.2 Time-dependent system reliability at time t under different τ

- In the second subplot with $\tau = 0.5$, the system reliability R_{sys} when $P = 1$ is most likely larger than when $P = 0$, and they both converge to 0.9992.
- In the third subplot with $\tau = 0.7$, the system reliability R_{sys} when $P = 1$ is always smaller than $P = 0$, and its maximum value when $P = 0$ is clearly larger than $P = 1$.

The system performance reacts differently at different values of τ . Smaller value results better performance given input $P = 1$, and larger value results higher reliable performance given input $P = 0$. To make the consistent system regardless of input patterns, medium value of τ is an option.

55.3.4 Optimal Stopping Time at Minimal Cost

It is impractical to process the system output for unlimited time; therefore, one of the primary goals is the budget control to make the model applicable. *Xie and Pham [15]* assumed that the error probabilities of each voter decrease as the time the voter spends on decision-making increases, which means that the system reliability is an increasing function of t . If t is limited, the system may not be reliable enough to guarantee the confidence for the decision. Thus, the motivation is to take the minimal time to generate an output with respect to higher reliability.

Linear Time-Dependent Cost Function

The goal is to determine the optimal stopping time to minimize the total cost by sacrificing limited amount of system

reliability. Define R_L as a prespecified least reliability value that the system is required to be achieved. Consider a simple linear time-dependent cost function,

$$LC(t) = c_1 (1 - R_t) + c_2 t \quad \text{such that } R_t \geq R_L \quad (55.7)$$

where c_1 is the cost associated with wrong decision and c_2 is the cost of unit time spent on decision process by all voters. In this case, both c_1 and c_2 are constants; however, they could also be time t dependent function as more complicated requirements may be considered in practice.

Denote $T = \operatorname{argmin}_{\{0 < t < \infty, R_t \geq R_L\}} LC(t)$ and T is the optimal stopping time that minimizes the cost $LC(t)$ subject to the constraint $R_t \geq R_L$ in limited time t . Then, define $t_0 = \inf \{t > 0 \mid R_t \geq R_L\}$ and t_0 is the minimum time to reach the prespecified value R_L . As R_t is a continuous and nondecreasing function of t , there exists an optimal time $T \in [t_0, t_0 + c_1 c_2^{-1} (1 - R_L)]$.

To obtain the numerical value of the optimal time T inside the interval $[t_0, t_0 + c_1 c_2^{-1} (1 - R_L)]$, one effective approach is to find the roots of the equation.

$$\left[\frac{d}{dt} LC(t) \right] \Big|_{t=T} = 0 \iff \left[\frac{d}{dt} R_t \right] \Big|_{t=T} = \frac{c_2}{c_1}$$

By solving it recursively from the *Newton-Raphson* algorithm, starting with an initial value $T^{(0)} \in [t_0, t_0 + c_1 c_2^{-1} (1 - R_L)]$, one can calculate the equation. For $k = 1, 2, \dots, m$.

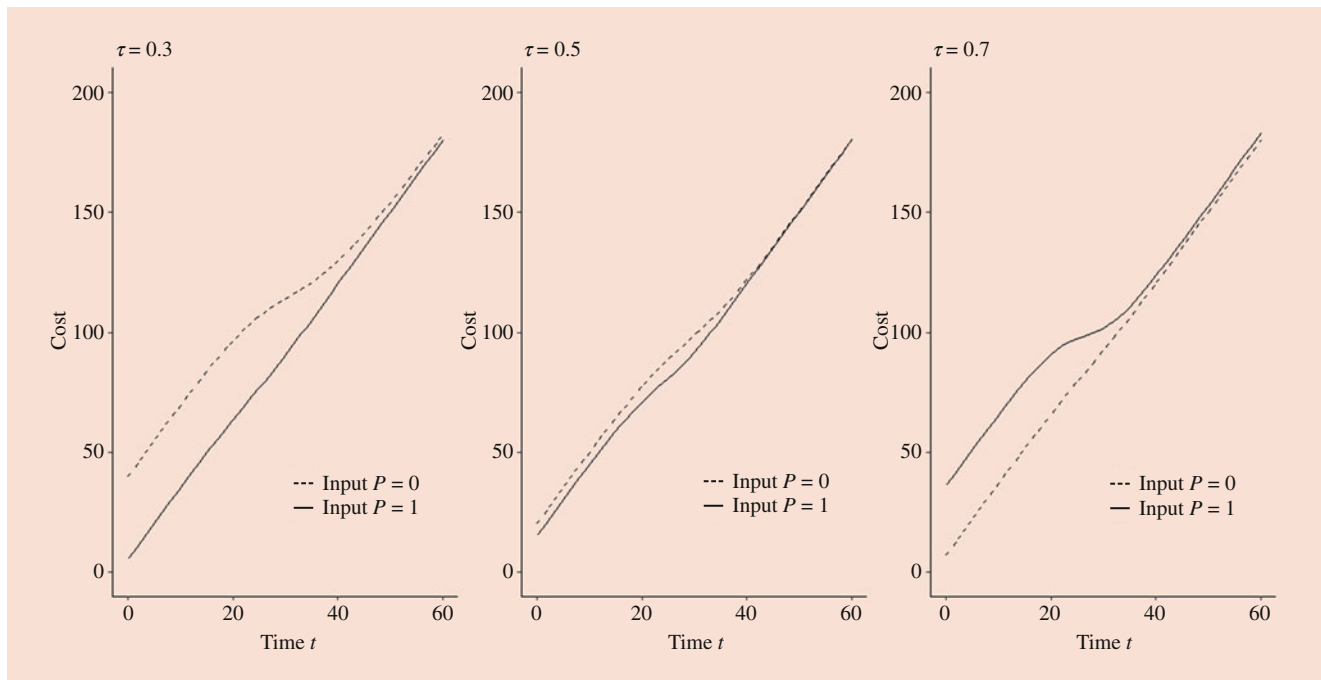


Fig. 55.3 Liner cost function at time t under different τ

$$T^{(k)} = T^{(k-1)} + \left\{ \left[\frac{d}{dt} R_t \right] \Big|_{t=T^{(k-1)}} - \frac{c_2}{c_1} \right\} \left[\frac{d}{dt} R_t \right] \Big|_{t=T^{(k-1)}} \quad (55.8)$$

As the algorithm is numerically convergent, we normally can get a root, T^* , for Eq. (55.8). If $T^* \in [t_0, t_0 + c_1 c_2^{-1} (1 - R_L)]$ and $C(T^*) < C(t_0)$, we obtain $T = T^*$, otherwise, $T = t_0$.

Example 55.5 Continuous with Example 55.4. Suppose $c_1 = 90$ and $c_2 = 3$. Figure 55.3 plots the cost functions $LC(t)$ against t for the six cases illustrated in Fig. 55.2.

Assume the least reliability value required is $R_L = 0.95$.

- In the case of $\tau = 0.3$, the minimum time to achieve R_L is $t_0 = 41$ if the input $P = 0$ and $t_0 = 17$ if the input $P = 1$.
- In the case of $\tau = 0.5$, the minimum time to achieve R_L is $t_0 = 35$ if the input $P = 0$ and $t_0 = 26$ if the input $P = 1$.
- In the case of $\tau = 0.7$, the minimum time to achieve R_L is $t_0 = 24$ if the input $P = 0$ and $t_0 = 37$ if the input $P = 1$.

Thus, according to the optimal time $T \in [t_0, t_0 + c_1 c_2^{-1} (1 - R_L)]$,

- If $\tau = 0.3$, $T \in [41, 42.5]$ if the input $P = 0$ and $T \in [17, 18.5]$ if the input $P = 1$.
- If $\tau = 0.5$, $T \in [35, 36.5]$ if the input $P = 0$ and $T \in [26, 27.5]$ if the input $P = 1$.
- If $\tau = 0.7$, $T \in [24, 25.5]$ if the input $P = 0$ and $T \in [37, 38.5]$ if the input $P = 1$.

By checking the monotonicity of the optimal time interval for all six cases, the optimal time T is,

- If $\tau = 0.3$, $T = 41$ if the input $P = 0$ and $T = 17$ if the input $P = 1$.
- If $\tau = 0.5$, $T = 35$ if the input $P = 0$ and $T = 26$ if the input $P = 1$.
- If $\tau = 0.7$, $T = 24$ if the input $P = 0$ and $T = 37$ if the input $P = 1$.

In other words, for instance, given $\tau = 0.5$, the system cost $LC(t)$ will be minimal while still achieving system reliability at least $R_L = 0.95$ at the time $T = 35$ if the input $P = 0$ and at the time $T = 26$ if the input $P = 1$.

Higher-Order Time-Dependent Cost Function

Linear time-dependent cost function provides a simple typical case; beside time cost and punishment from wrong decision, there could be more cost such as labor cost and capital cost that depends on specific requirements. On the other hand, higher-order time-dependent cost function may also be applied on some situations when different costs are considered unequally weighted.

Consider a quadric time-dependent cost function with nonlinear cost on unit time,

$$HC(t) = c_3 (1 - R_t) + c_4 t^2 \quad \text{such that } R_t \geq R_L \quad (55.9)$$

where c_3 is the cost associated with wrong decision and c_4 is the cost of square of unit time by all voters. In this case, extra

time spent during the process of decision-making is more expensive to boost the cost out of the budget.

Similarly, define $T = \operatorname{argmin}_{\{0 < t < \infty, R_t \geq R_L\}} \text{HC}(t)$ be the optimal stopping time that minimizes the cost $\text{HC}(t)$ subject to the constraint $R_t \geq R_L$ in limited time t . Then, $t_0 = \inf \{R_t \geq R_L | t > 0\}$ is the minimum time to reach the prespecified value R_L . As R_t is a continuous and nondecreasing function of t , one can find an optimal time $T \in \left[\sqrt{t_0}, \sqrt{t_0 + c_3 c_4^{-1} (1 - R_L)} \right]$.

To obtain the optimal time T inside the interval $\left[\sqrt{t_0}, \sqrt{t_0 + c_3 c_4^{-1} (1 - R_L)} \right]$, the same approach yields the following equation.

$$\left[\frac{d}{dt^2} \text{HC}(t) \right] \Big|_{t=T} = 0 \iff \left[\frac{d}{dt^2} R_t \right] \Big|_{t=T} = \frac{2c_4}{c_3}$$

Applying *Newton-Raphson* algorithm with an initial value $T^{(0)} \in \left[\sqrt{t_0}, \sqrt{t_0 + c_3 c_4^{-1} (1 - R_L)} \right]$, one can calculate the equation. For $k = 1, 2, \dots, m$.

$$T^{(k)} = T^{(k-1)} + \left\{ \left[\frac{d}{dt^2} R_t \right] \Big|_{t=T^{(k-1)}} - \frac{2c_4}{c_3} \right\} \left[\frac{d}{dt^2} R_t \right] \Big|_{t=T^{(k-1)}}$$

Then, if $T^* \in \left[\sqrt{t_0}, \sqrt{t_0 + c_3 c_4^{-1} (1 - R_L)} \right]$ and $C(T^*) < C(t_0)$, we obtain $T = T^*$, otherwise $T = t_0$.

Example 55.6 Continuous with Example 55.5. To better depict the difference from linear case, assume $c_3 = 90$ and $c_4 = 3$ equivalent to c_1 and c_2 . Figure 55.4 plots the cost functions $\text{HC}(t)$ against t for the six cases illustrated in Fig. 55.2. And the comparison can be illustrated clearly from Fig. 55.3.

Assume the least reliability value required is $R_L = 0.95$; as the time-dependent reliability function preserves, one can obtain the identical minimum time to achieve R_L for input $P = 0$ and $P = 1$, respectively.

- In the case of $\tau = 0.3$, the minimum time to achieve R_L is $t_0 = 41$ if the input $P = 0$ and $t_0 = 17$ if the input $P = 1$.
- In the case of $\tau = 0.5$, the minimum time to achieve R_L is $t_0 = 35$ if the input $P = 0$ and $t_0 = 26$ if the input $P = 1$.
- In the case of $\tau = 0.7$, the minimum time to achieve R_L is $t_0 = 24$ if the input $P = 0$ and $t_0 = 37$ if the input $P = 1$.

Thus, according to the optimal time $T \in \left[\sqrt{t_0}, \sqrt{t_0 + c_3 c_4^{-1} (1 - R_L)} \right]$,

- If $\tau = 0.3$, $T \in [6.40, 6.52]$ if the input $P = 0$ and $T \in [4.12, 4.30]$ if the input $P = 1$.
- If $\tau = 0.5$, $T \in [5.92, 6.04]$ if the input $P = 0$ and $T \in [5.09, 5.24]$ if the input $P = 1$.
- If $\tau = 0.7$, $T \in [4.90, 5.05]$ if the input $P = 0$ and $T \in [6.08, 6.21]$ if the input $P = 1$.

By checking the monotonicity of the optimal time interval for all six cases, the optimal time T is,

- If $\tau = 0.3$, $T = 6.40$ if the input $P = 0$ and $T = 4.12$ if the input $P = 1$.
- If $\tau = 0.5$, $T = 5.92$ if the input $P = 0$ and $T = 5.09$ if the input $P = 1$.
- If $\tau = 0.7$, $T = 4.90$ if the input $P = 0$ and $T = 6.08$ if the input $P = 1$.

As expected, cost of unit time in quadric formula reaches to a high level much faster than linear formula. It is much harder to keep the system reliable at this case since there is much less time in the process of decision-making. From the comparison of Examples 55.5 and 55.6, one can tell the budget control substantially influences decision-making.

55.4 Weighted Threshold Indecisive Voting System

Although the situation of no decision of any voter has been noticed in the basic model assumptions, the weighted voting system introduced in Sect. 55.3 uses a classic decision rule without considering indecisive effect, that is, weights of voters who have no decision do not get involved in calculations. This will likely cause bias even in some simple situations. For example, suppose an HO system with 10 voters makes the decision on a proposal. Eight voters have no decision, and the other two have decisions with one acceptance and one rejection. If the one-vote acceptance has higher weights than the one-vote rejection, the final decision is acceptance according to the classic decision rule. However, this may not be true as the summation of weights for 8 voters with no decision may be much larger, and the proper decision might be either rejection or reevaluation.

Zhang and Pham [16] studied a generalized weighted indecisive voting n -units system using a new decision rule considering indecisive effect, where the classic decision rule becomes the special case of the indecisive decision rule. They are trying to scale the effects of indecisiveness to the entire system performance in terms of system reliability. The numeric examples do the comparison between two decision rules and illustrate the sensitivities of the model.

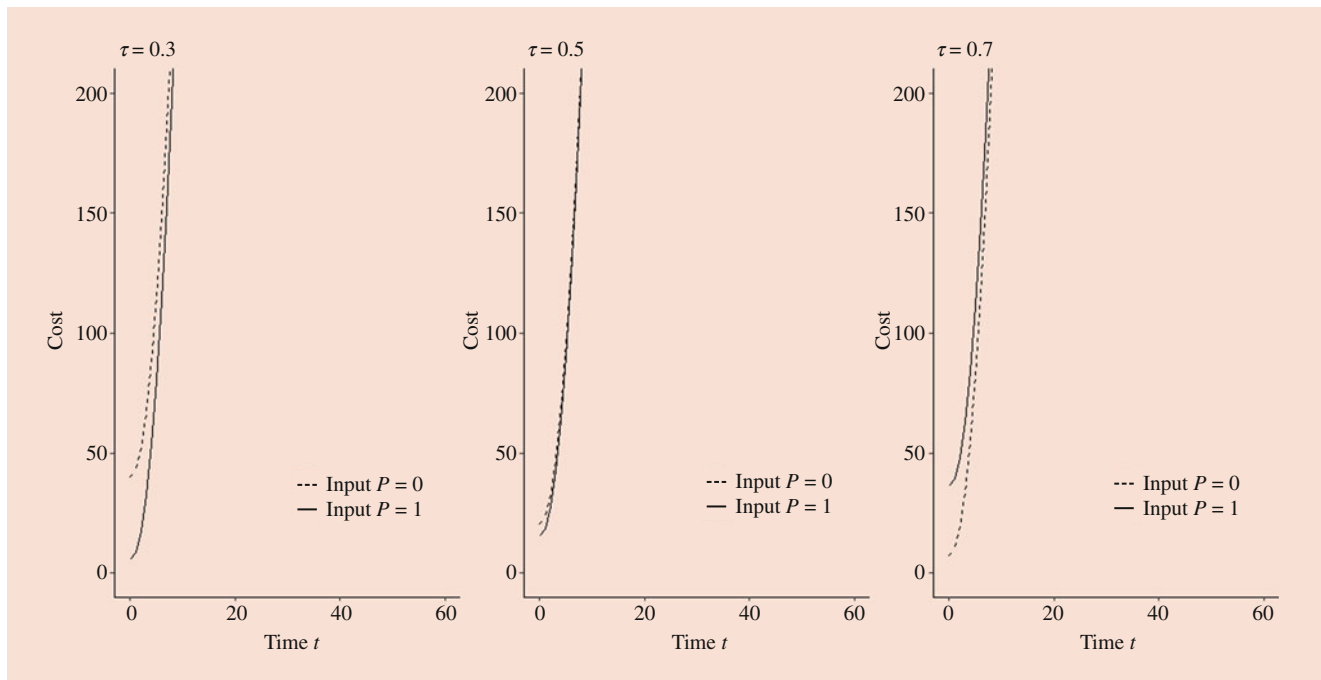


Fig. 55.4 Quadric cost function at time t under different τ

55.4.1 Model Assumptions

The basic model assumptions are the same as ones defined in Sect. 55.3 except for the last one, the law of the decision rule. In this section, a decision rule is designed with two prespecified parameters of threshold and indecision (τ, θ) incorporating all subdecisions to generate the final decision. The additional indecision parameter, θ , is introduced in the decision rule as indecisive effect is considered. Note that θ has domain different than τ as θ can be zero, which makes the indecisive decision rule be the classic one described in Sect. 55.3.

55.4.2 Indecisive Decision Rule

The weights of indecisive voters are treated as obstacles of decision-making because no decision is likely to be considered a negative decision. Given any input, a decision should be made as acceptance if the input is favorable and vice versa. In such a case, no decision is more likely a nonfavorable decision, which may result a wrong final decision or reevaluation. Thus, given the input is 1 (favorable), the system output is 1 if and only if the follow equation is satisfied.

$$\sum_{i \in I_1} w_i \geq \tau \left(\sum_{i \notin I_x} w_i + \theta \sum_{i \in I_x} w_i \right) \quad (55.9)$$

Total weights of each voter who votes 1 in the left-hand side need to exceed partial of the total weights where the indecisive weights are drained out accordingly. A parameter θ used to distinguish no decision from wrong decision is applied on the indecisive weights in the right-hand side. Note θ can be zero, which results the decision rule to be the classic one in Sect. 55.3.

55.4.3 Indecisive Model Formulation

To formulate the indecisive model, Zhang and Pham [16] regenerate the recursive function to avoid imposing restrictions [11, 14]. They used the same technic to define a random variable X_i where $i = 1, 2, \dots, n$ where indecisive index is newly introduced.

$$X_i = \begin{cases} 1 - \tau & \text{if } i \in I_1 \\ -\tau & \text{if } i \in I_x \\ -\tau\theta & \text{if } i \in I_0 \end{cases} \quad (55.10)$$

Rewriting Eq. (55.10), the system output is 1 (favorable) if and only if,

$$\sum_{i \in I_1} w_i - \tau \sum_{i \notin I_x} w_i - \tau\theta \sum_{i \in I_x} w_i \geq 0 \quad (55.11)$$

Simplifying (55.11) for a clear vision, the second sum can be disaggregated and then rearranged.

$$\sum_{i \in I_1} (1 - \tau) w_i - \sum_{i \in I_0} \tau w_i - \sum_{i \in I_x} \tau \theta w_i \geq 0 \tag{55.12}$$

$$\iff \sum_{i=1}^n w_i X_i \geq 0$$

Equation (55.12) clearly interprets contribution values of three types of individual decision, where the positive value $(1 - \tau)w_i$ of voter i accumulates if it votes acceptance, the negative value $-\tau w_i$ of voter i summed up if it votes rejection, and the negative value $-\tau \theta w_i$ of voter i is added together if it abstains. Then, two similar probability functions can be induced as follows.

$$Q_n(s) = \Pr\left(\sum_{i=1}^n w_i X_i \geq s \mid P = 1\right)$$

$$\tilde{Q}_n(s) = \Pr\left(\sum_{i=1}^n w_i X_i < s \mid P = 0\right)$$

Initializing variable $s = 0$, one obtains conditional probability functions,

$$Q_n(0) = \Pr\left(\sum_{i=1}^n w_i X_i \geq 0 \mid P = 1\right) = \Pr(S = 1 \mid P = 1)$$

$$\tilde{Q}_n(0) = \Pr\left(\sum_{i=1}^n w_i X_i < 0 \mid P = 0\right) = \Pr(S = 0 \mid P = 0)$$

Hence, the reliability of the weighted threshold indecisive voting system with n voters is,

$$R_{\text{sys}} = Q_n(0) \Pr(P = 1) + \tilde{Q}_n(0) \Pr(P = 0)$$

where $Q_n(s)$ and $\tilde{Q}_n(s)$ can be written in terms of recursive functions, and the restricted domains are different from the classic ones illustrated in Sect. 55.3. Simply, an indecisive parameter θ is considered showing as follows,

For $n \geq 1$,

$$Q_1(s) = \begin{cases} 1 & \text{if } s \leq -\tau w_1 \\ p_{1,11} + p_{1,1x} & \text{if } -\tau w_1 < s \leq -\tau \theta w_1 \\ p_{1,11} & \text{if } -\tau \theta w_1 < s \leq (1 - \tau) w_1 \\ 0 & \text{if } s > (1 - \tau) w_1 \end{cases}$$

$$\tilde{Q}_1(s) = \begin{cases} 0 & \text{if } s \leq -\tau w_1 \\ p_{1,00} & \text{if } -\tau w_1 < s \leq -\tau \theta w_1 \\ p_{1,00} + p_{1,0x} & \text{if } -\tau \theta w_1 < s \leq (1 - \tau) w_1 \\ 1 & \text{if } s > (1 - \tau) w_1 \end{cases}$$

For $n \geq 2$

$$Q_n(s) = Q_{n-1}(s_n^1) p_{n,11} + Q_{n-1}(s_n^x) p_{n,1x} + Q_{n-1}(s_n^0) p_{n,10}$$

$$\tilde{Q}_n(s) = \tilde{Q}_{n-1}(s_n^1) p_{n,11} + \tilde{Q}_{n-1}(s_n^x) p_{n,1x} + \tilde{Q}_{n-1}(s_n^0) p_{n,10}$$

where $s_n^1 = s - (1 - \tau) w_n$, $s_n^x = s + \tau w_n$, and $s_n^0 = s + \tau \theta w_n$. One more parameter from indecisive effect recursively makes the equations more complicated as the n goes up. Two examples are provided varying values for one parameter while another one is fixed, and it aims to provide a brief knowledge of an across relation between two parameters and a simple comparison between two decision rules.

55.4.4 Numerical Example of Indecisive System Reliability

Example 55.7 Consider a grant institution votes to fund good proposals and suppose the voting group has $n = 4$ members and probabilities of their judgment errors with assigned weights are given the same as in Example 55.2. Fix the indecisive parameter $\theta = 0$; the calculations from three values of $\tau = \{0.1, 0.5, 0.9\}$ are tabled, where weights are normalized. Note that $\theta = 0$ makes the indecisive decision rule identical in Sect. 55.3. This example shows how single parameter τ is affecting the system reliability.

In Table 55.2, system reliability under five different prior distributions of inputs P are calculated. Two extreme cases of threshold τ in third and fifth rows indicate inverse results, where the system is more reliable of favorable proposals when τ is extremely small and vice versa when τ is extremely high. From the calculations, the system guarantees the quality of proposals in the way that bad proposals are easily rejected, and good proposals seem to be accepted competitively. Figure 55.5 intuitively summaries the trend of performance in terms of system reliability.

The system reaches a steady state regardless of the quality of proposals when $\tau = 0.5$. The initialization of τ really depends on how much knowledge of the input is obtained. This example recalls the classic decision by setting $\theta = 0$ and extends the calculations by illustrating two more extreme cases of τ .

Example 55.8 (Extended Example 55.7) Calculations of three fixed values of $\tau = \{0.1, 0.5, 0.9\}$ while the values of θ vary. This example shows how cross effect influences the system reliability from both parameters.

- A higher value of θ downgrades R_{sys} given favorable proposals; however, it increases R_{sys} given nonfavorable proposals. Given τ , one can see larger variations of R_{sys} for

Table 55.2 System reliability with $\theta = 0$

Pr($P = 1$)	1.0	0.8	0.5	0.2	0.0
τ	R_{sys} with $\theta = 0$				
$\tau = 0.1$	0.9956	0.9108	0.7836	0.6564	0.5717
$\tau = 0.5$	0.9214	0.9207	0.9196	0.9186	0.9178
$\tau = 0.9$	0.5400	0.6311	0.7678	0.9045	0.9956

favorable proposals than nonfavorable ones with respect to θ .

- In practice, high penalties for indecisive decisions result in more competitive situations for favorable proposals. In

the meanwhile, this can easily prevent the acceptance from nonfavorable proposals since indecisive votes are likely treated as nonacceptable votes.

It turns out that indecisive effect is sensitive in types of inputs. Figure 55.6 visualizes Table 55.3 by showing the trend of performance via different initializations of θ in terms of system reliability under different distributions of inputs.

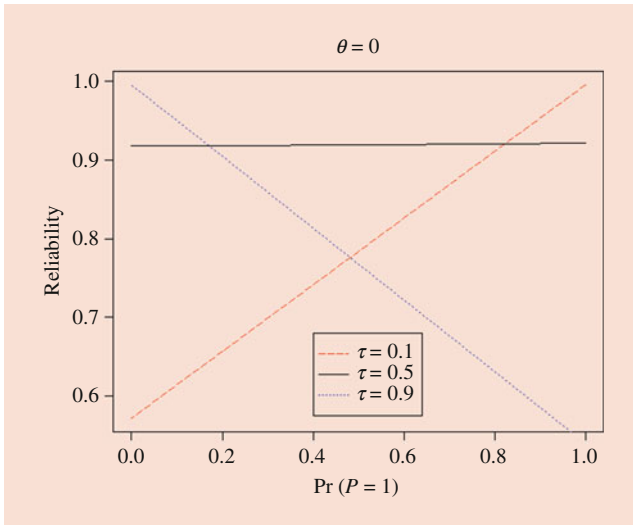


Fig. 55.5 Illustration of system performance in Table 2

- Higher probabilities of favorable proposals result in larger R_{sys} with smaller τ when θ is fixed, and with smaller θ if τ is fixed. The situation overturns with unfavorable proposals. Somehow, it indicates the sensitivity of the prior distribution of inputs.
- Both high values of τ when θ is fixed and high values of θ when τ is fixed make favorable proposals competitively acceptable and nonfavorable ones easily rejected. Thus, initializations of both parameters affect the system performance, and one has the independent theory to optimize the system associated with prespecified assumptions.
- Overall, the system is stable under $\tau = 0.5$ whenever θ changes, which provides a naive option when prior information of input is limited.

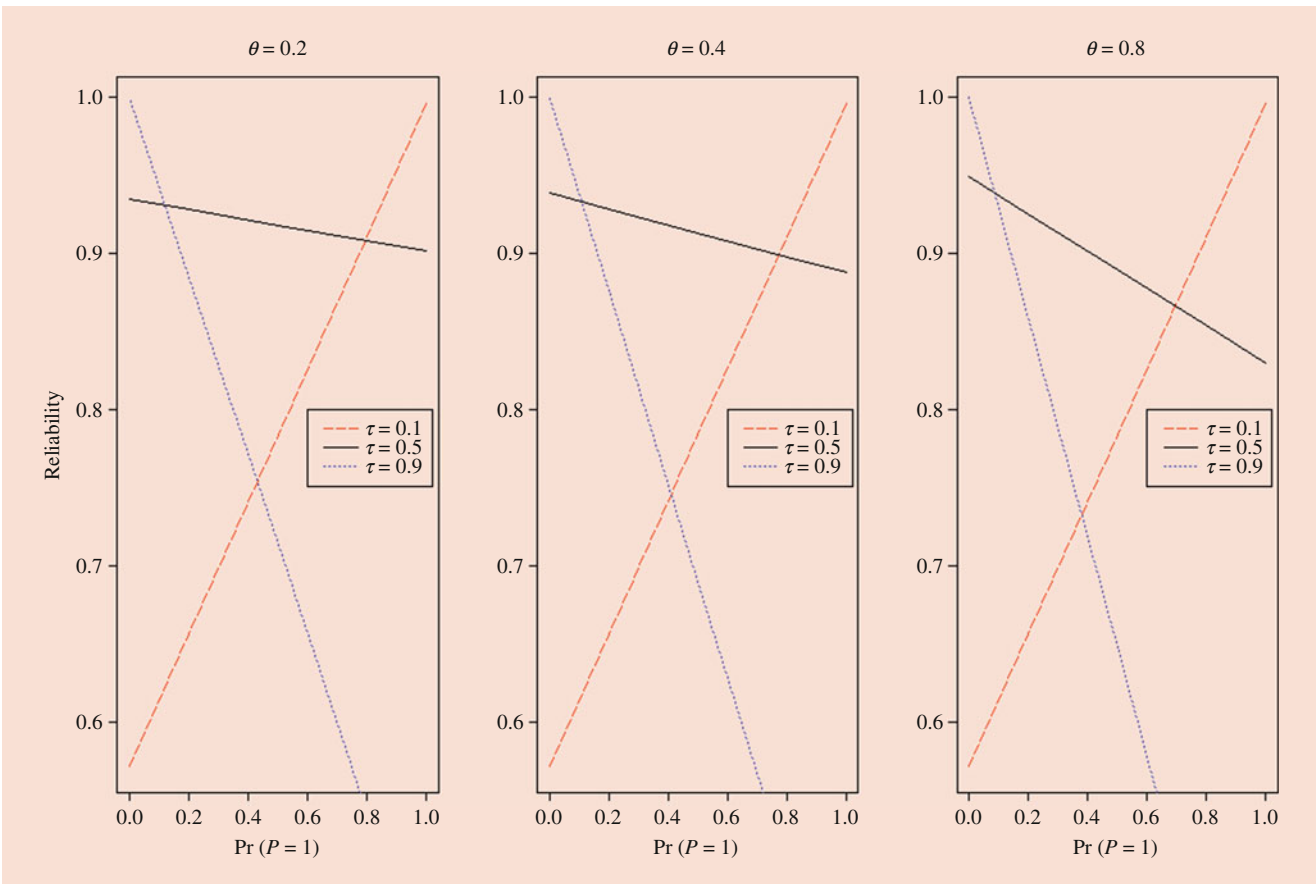


Fig. 55.6 Illustration of system performance in Table 55.3

Table 55.3 System reliability with $\tau = \{0.1, 0.5, 0.9\}$

$\Pr(P = 1)$	1.0	0.8	0.5	0.2	0.0
θ	R_{sys} with $\tau = 0.1$				
$\theta = 0.0$	0.9956	0.9108	0.7836	0.6564	0.5717
$\theta = 0.2$	0.9954	0.9107	0.7836	0.6566	0.5719
$\theta = 0.4$	0.9954	0.9107	0.7836	0.6566	0.5719
$\theta = 0.8$	0.9954	0.9107	0.7836	0.6566	0.5719
θ	R_{sys} with $\tau = 0.5$				
$\theta = 0.0$	0.9214	0.9207	0.9196	0.9186	0.9178
$\theta = 0.2$	0.9011	0.9078	0.9178	0.9279	0.9346
$\theta = 0.4$	0.8876	0.8977	0.9130	0.9282	0.9383
$\theta = 0.8$	0.8298	0.8537	0.8894	0.9251	0.9489
θ	R_{sys} with $\tau = 0.9$				
$\theta = 0.0$	0.5400	0.6311	0.7678	0.9045	0.9956
$\theta = 0.2$	0.4279	0.5421	0.7134	0.8846	0.9988
$\theta = 0.4$	0.3820	0.5054	0.6905	0.8756	0.9990
$\theta = 0.8$	0.2948	0.4358	0.6473	0.8587	0.9997

- Nonzero θ downgrades the overall system performance. It is straightforward when threshold τ is large by comparing values of intersection of x -axis representing input P .

In real applications, the threshold value is very substantial in deterministic optimization of the system performance. Given the size of the voters and the error probabilities of each voter with respect to each type of individual decision, cross effects from both parameters θ and τ bring complications in decision-making along with uncertain types of inputs. As discussed in Sect. 55.3, cost function is time-dependent; to have a better understanding of prior information of inputs, certain effort is required. However, a budget constraint would then change the optimization process as extra restrictions relocate the local optimal.

55.4.5 Sensitivity Analysis on WVS

Two major reasons drive the test of different combinations of (τ, θ) along with different qualities of voters. First, system reliability is used to quantify system reactions under certain policies and constraints including maximal system reliability or minimal budget cost, therefore, resulting in location detections of parameterization. Moreover, the decision process becomes less optimistic when the prior information is limited, which again relates to the assumptions of budget constraints. What is interesting is the mutual interactions between them and makes them become a closed loop. To loop out of the circle, sensitivity analysis may help to locate the equilibrium.

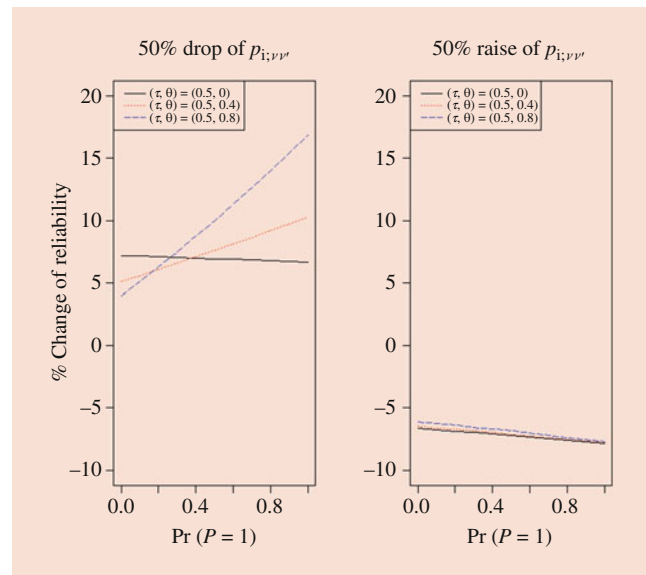


Fig. 55.7 Sensitivity on failure rate

Intrinsic Sensitivity

Example 55.9

(Sensitivity on Failure Rate of Voters) Recall Example 55.2 as a control case. Zhang and Pham [16] showed the 10% decrease/increase of all types of errors of each voter has limited influence on system reliability, which outstands the advantage of the model design. In this example, 50% decrease/increase of failure rate of each voter is applied on their model. Figure 55.7 plots the change of system reliability at $\theta = \{0.0, 0.4, 0.8\}$ with $\tau = 0.5$ while other variables are fixed. Table 55.4 compares two cases.

Different from 10% change of failure rate [16], 50% change of failure rate has significant distinctions in drop and raise failure rates.

- First, 50% drop/raise of $p_{i,vv'}$ (reliable/less reliable voters) results a 5–15%/5–10% increase/decrease in system reliability among all three cases.
- Moreover, the variation of system reliability in drop case is much bigger than in the raise case, which is different from 10% change of failure rate.
- The raise case has more advantages in real applications. Much less reliable voters result limited downgrade in system performance. And variations of the system reliability are small enough among three combinations of parameterizations.

Table 55.4 Reliability of different $p_{i,vv'}$ with $(\tau, \theta) = (0.5, 0.0)$

		R_{sys}		
$\Pr(P=1)$	Control	10% Drop $p_{i,vv'}$	10% Raise $p_{i,vv'}$	Change in %
0.0	0.9178	0.9339	0.9000	1.75 / -1.94
0.4	0.9193	0.9351	0.9017	1.72 / -1.91
0.8	0.9207	0.9363	0.9033	1.69 / -1.89
1.0	0.9214	0.9369	0.9041	1.68 / -1.88
		R_{sys}		
$\Pr(P=1)$	Control	50% drop $p_{i,vv'}$	50% raise $p_{i,vv'}$	Change in %
0.0	0.9178	0.9838	0.8571	7.19 / -6.61
0.4	0.9193	0.9835	0.8540	6.98 / -7.10
0.8	0.9207	0.9831	0.8508	6.78 / -7.59
1.0	0.9214	0.9829	0.8492	6.67 / -7.84

Table 55.4 exhibits the calculated numbers for both 10% and 50% cases with $(\tau, \theta) = (0.5, 0)$, which shows 5 times difference in voters' qualities results less than 5 times difference in system reliability. This may indicate the advantage of the system design.

Example 55.10

(Sensitivity on τ) Recall Example 55.2 as a control case. This example focuses on how τ affects the system by fixing all other variables and parameters. Five different values of $\tau = \{0.1, 0.3, 0.5, 0.7, 0.9\}$ are applied in calculations. Figure 55.8 plots three cases given $\theta = \{0.0, 0.4, 0.8\}$. First subplots of Fig. 55.8 represent the classic decision rule as indecisiveness is excluded with $\theta = 0$.

- When $\tau < 0.5$, system reliability R_{sys} has an increasing trend as the system is reliable given favorable proposals. When $\tau > 0.5$, system reliability R_{sys} has a decreasing trend as the system is less reliable given favorable proposals.
- $\tau = 0.5$ makes the system reach to a steady state given any inputs if indecisiveness is not considered ($\theta = 0$); however, this is not the case when indecisiveness is applied in decision rule as one can see in the second and third subplots of Fig. 55.8.
- Indecisiveness makes the system more sensitive in choices of threshold τ . The overall system performance is reliable as $\min_P R_{sys} > 0.5$ independent from the choices of τ in the first subplot of Fig. 55.8, but the system become less stable with large value of τ when $\theta \neq 0$.

Example 55.10

(Sensitivity on θ) Recall Example 55.2 as a control case. This example focuses on how θ affects the system by fixing all other variables and parameters. Five different values

of $\theta = \{0.1, 0.3, 0.5, 0.7, 0.9\}$ are applied in calculations. Figure 55.9 plots three cases given $\tau = \{0.1, 0.5, 0.9\}$. In this example, $\theta = 0$ is omitted as it studies θ influence. One can see the domination of τ as it determines how system is affected in terms of R_{sys} , although θ indeed has internal effects.

- When $\tau = 0.1$, all 5 lines representing different options of θ overlap together. θ has no influence at all. As the threshold is small, it makes indecisiveness small enough to lose power to play in the role. In fact, this also proves small threshold is less practical.
- As τ gets bigger, θ is brought back to the game. From both second and third subplots of Fig. 55.9, larger θ makes the system less reliable if τ is fixed. However, θ causes larger variations when τ is large; therewith, θ is sensitive on condition of τ .
- When $\tau = 0.5$, system is stable even with the missing information of inputs. Although the system is less sensitive on θ , it can still be a secondary factor in model modifications as τ is the dominant factor in decision rule.

Extrinsic Sensitivity

The indecisive effect can be applied when people have difficulties in making decisions. If more time t can be spent to collect more information, it may be helpful to make decisions. In fact, this is meaningful in social activities. Zhang and Pham [16] have done some jobs on this problem. Three extreme conditions are considered unless any specific motivation is specified. Without loss of generality, more information is assumed to decrease the probability of no decision $p_{i,vx}$ where $v = \{0, 1\}$ for voter i and $x \in v'$. However, one does not know if extra information contributes either positively or negatively. Therefore, it would be wise to assume that decreasing values have three different flows.

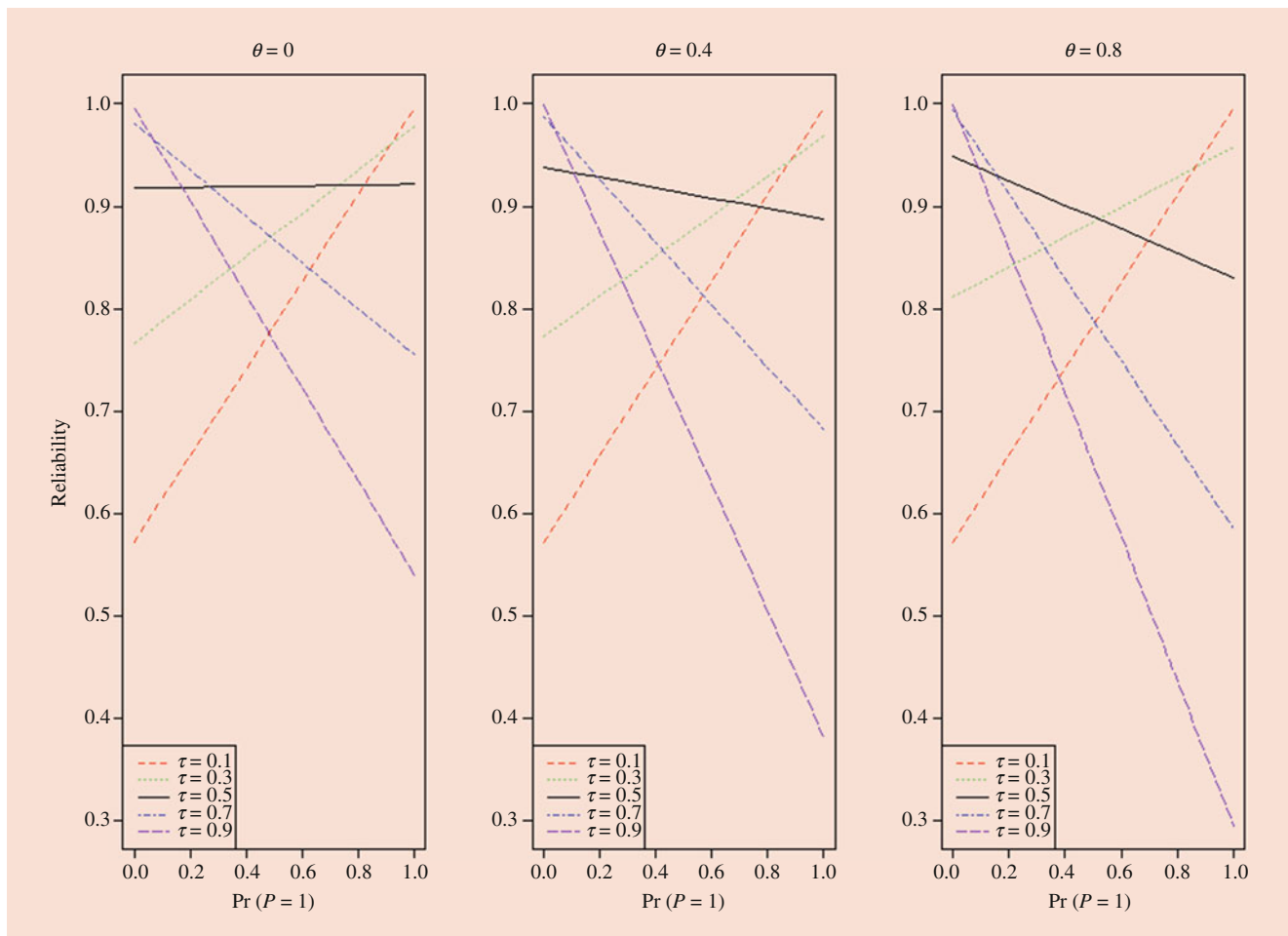


Fig. 55.8 Sensitivity on τ

- *Flow 1*: Decreasing values $\tilde{p}_{i,vx}$ from indecisive probabilities all contribute to probability of making correct decision, that is, $p_{i,00} = p_{i,00} + \tilde{p}_{i,0x}$ and $p_{i,11} = p_{i,11} + \tilde{p}_{i,1x}$.
- *Flow 2*: Decreasing values $\tilde{p}_{i,vx}$ from indecisive probabilities all contribute to probability of making wrong decision, that is, $p_{i,01} = p_{i,01} + \tilde{p}_{i,0x}$ and $p_{i,10} = p_{i,10} + \tilde{p}_{i,1x}$.
- *Flow 3*: Decreasing values $\tilde{p}_{i,vx}$ from indecisive probabilities contribute to probability of making correct and wrong decision with equal probability, that is, $p_{i,00} = p_{i,00} + 0.5\tilde{p}_{i,0x}$ and $p_{i,01} = p_{i,01} + 0.5\tilde{p}_{i,0x}$ if input $P = 0$, and $p_{i,10} = p_{i,10} + 0.5\tilde{p}_{i,1x}$ and $p_{i,11} = p_{i,11} + 0.5\tilde{p}_{i,1x}$ if input $P = 1$.

Example 55.11

(Sensitivity on time T) After certain effort made by time T , voters' probability of indecision drops to half of the original level, $\tilde{p}_{i,vx} = 0.5p_{i,vx}$, where $p_{i,vx}$ is from Example 55.2 as a control experiment. To minimize the side effect from initialization of two parameters, $\tau = 0.5$ is applied. Figure 55.10 depicts R_{sys} for all three flows under three cases of $\theta = \{0.1, 0.5, 0.9\}$.

- The solid black curve on each subplot represents the control experiment for all three cases. Flow 1 and flow 2 represent two extreme conditions which define the upper and lower bounds for the black curve.
- The lower bound, flow 2, is the worst condition that indicates the waste of the effort as system becomes less reliable. In fact, this is less likely to happen.
- The ideal condition of flow 1 indicates certain effort is worthy as system reliability improves quite a bit, and flow 3 is showing some credits of the effort from the third subplots.
- This example indeed gives confidence on effort of lowering indecisive effect, and applying some risk management will likely control the direction above flow 3 and reach to flow 1 as possible.

One can also consider this problem associated with cost control as the discussion in Sect. 55.3. In this example, a constant function after time T is studied. However, one can still apply a time-dependent function if restrictions only allow $t < T$ spent on effort as it is more practical in real applications.

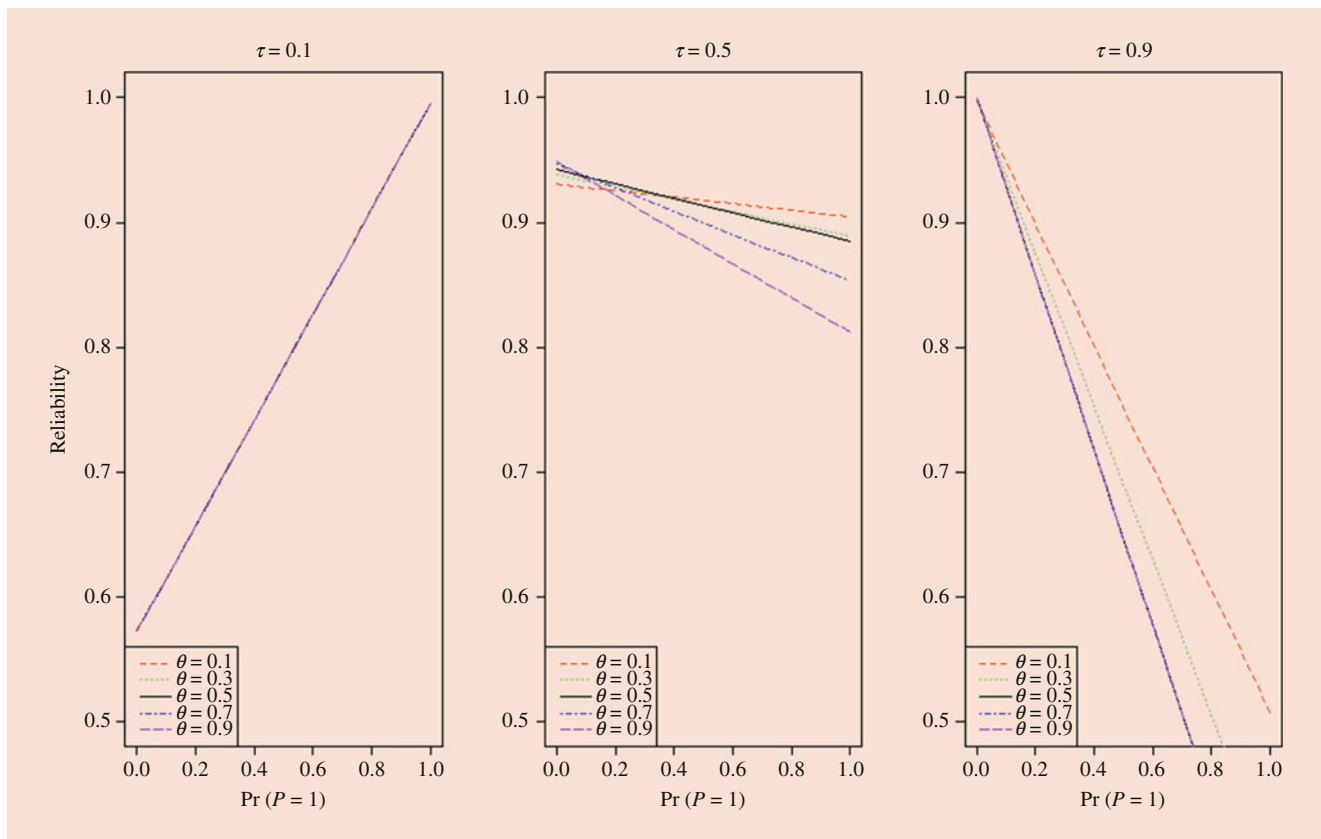


Fig. 55.9 Sensitivity on θ

55.5 Other Related Works

55.5.1 Model-Related Studies

Based on the framework of weighted voting systems, some key components of WVS are studied. And those ideas and techniques are critical and creative to reach the goal of optimizations under various practical requirements.

Levitin [19] showed the additional reliability improvement achieved by grouping units in voting subsystems and tallying the weighted votes of such subsystems to make a final decision. He aimed to answer the question that if there was a way to make WVS more reliable without affecting the failure rate of individual voter, which could be achieved by grouping voters into several separate voting subsystems sharing the same decision rule. Each category was considered as a new united voter with new weights generated from each element of the category until final decision was made; therefore, the threshold value τ needed to be further determined into a group of threshold values with respect to each group. A specific algorithm was suggested that obtains the optimal element grouping as well as weights within each group, and its evaluation process was based on universal generating techniques. He concluded that unit grouping is most beneficial

when some units had very similar probabilities of making correct decisions to certain type of inputs.

Long et al. [20] studied the system reliability optimization of WVS with continuous states inputs, where they derived the analytical expression. For the reliability of the entire system under certain distribution assumptions. They pointed out the limitation of many existing models with inputs having very small state spaces. Furthermore, the number of different combinations of generated output rose dramatically with increased input states. Therewith, they provided an example to describe the configuration of a WVS consisting of n independent voters with continuous states input. To evaluate the effectiveness and accuracy of the model, they applied Monte Carlo Simulations to reach the randomization requirement to quantify the model. Some optimization problems with cost constraints were formulated based on continuous inputs, where they concluded that different types of voters resulted different allocations under various cost constraints.

Foroughi and Aouni [21] applied data envelopment analysis to show the drawbacks of the assignment of weights in the existing model and create an approach for determining a common set of weights of voters in a WVS. Weights obtained from inefficient voters and their sensitivity to the votes from the worst voters were major shortcomings needed to be overcome. They built a new model using only the

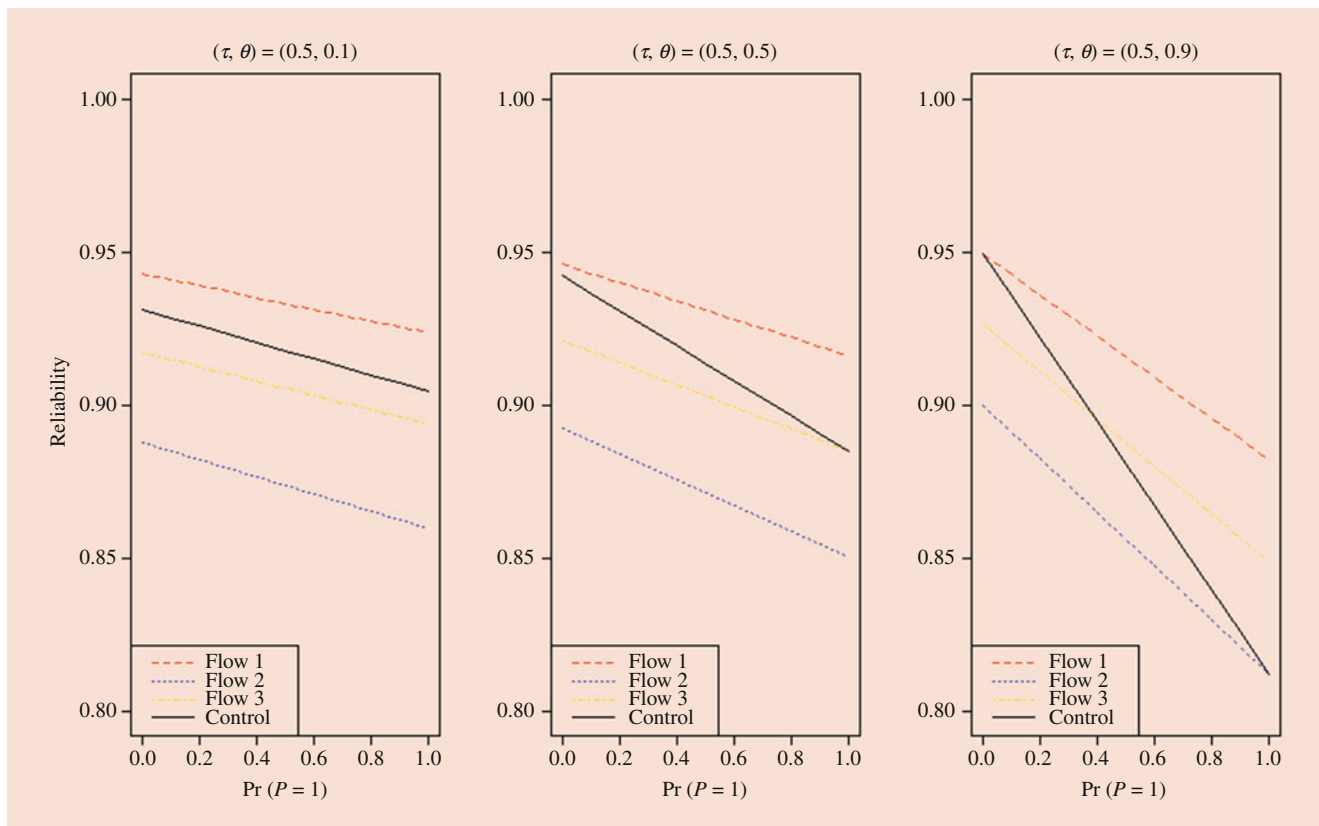


Fig. 55.10 Sensitivity on T

information about efficient voter and prevented selection of zero weights. Moreover, to enlarge the discrimination power and minimize the possibility of having multiple optimal solutions, some additional objective functions were used. Examples of a new generated two-objective function model are utilized as a combination of the new model they built with the existing model. They addressed some fundamental but significant works on weights of a WVS, which is crucial to quantify the system performance. More research needs to be done on determination of weights of voters, one of the key components of a WVS.

55.5.2 Application-Related Studies

Besides traditional applications of WVS on target detection, communication systems, and pattern recognition, WVS can be applied on more advanced modern applications as the characteristics of WVS are well established so far. Case studies such as works mentioned below are indicating accuracy and effectiveness of weighted voting systems.

Sun and Han [22] proposed a dynamic weighted voting based on knowledge discovery on multiple classifier systems to improve its performance. Principal components analysis was used for the feature extraction and decorrelation

to reduce the pattern dimensions. Then, the model of the new classifier fusion was built by introducing the algorithm step by step, and experimental results indicate the system performance of the classification can be improved; therefore, the proposed method in image classification is effective. They concluded two attractive characteristics including operation of base classifiers and auto-generation of mass function induced from interclass in different feature spaces. In such a way, weights assigned to each output of the base classifier were able to estimate by the separability of the training sample sets in relevant feature spaces, and all weights were able to be calculated by a modified heuristic fusion function and assigned randomly to each classifier varying with its output from mass functions.

Levitin [23] studied a WVS of units with nonsimultaneous outputs delivery, where the decision time of the entire system was dependent on the distribution of unit weights and on the value of threshold in decision rule. Decision-making on a WVS in practice usually had constraint of limited time; therefore, a measurement of its performance was considered as the expected system decision time. He formulated the problem of the optimal system equivalent to locate voters' weights and threshold that maximize the system reliability while providing the expected decision time less than a predetermined value. The algorithm based on universal generating function

for evaluating WVS reliability and expected decision time illustrated the existence of the trade-off between the system reliability and its rapidity.

Liu and Zhang [24] built a WVS model with two different kinds of unreliable links including type I of two states with reliable connection and complete failure and type II of three states with reliable connection, error, and failure. Outputs of voting units might not be properly transferred to a sink unit because of the unreliability of links between voting units and sink units. Wireless sensor network was applied in their model design and formulation. By illustrating examples, they studied the influence of a specified link's reliability and some system variables on reliability optimization on a WVS. Under type I links, the system was more reliable with the increase of link quality or the number of units. If type II links were applied, the effect from the change on the mean value of unit reliability was reduced with the increasing number of voting units. This work focused on a potential component of WVS that could be significant but easily ignored.

55.6 Conclusion and Future Research

In this chapter, the framework of weighted voting system with nonidentical voters subject to three types of failure rate is presented. For the weighted threshold voting system, three key components including *voters*, *weights*, and *decision rules* are introduced. Two decision rules are considered. One is a classic decision rule without indecisive effect, and the other one is an extended decision rule considering indecisive effect. System reliability is used to quantify the system performance of a WVS, and some approximation techniques are applied under certain purposes. Numeric examples are illustrated to better understand how WVS evaluate the whole process of decision-making. Sensitivity analysis is presented via different optimization problems with certain constraints. Last but not least, many related works that cannot be fully explained are also mentioned to give a brief but full picture of current results of WVS.

The results of indecisive effect along with recursive evaluation functions presented in this chapter are a recent research. Although sensitivity analysis shows its potential of widely usage in practice, some new problems are also induced such as the new indecisive parameter in decision rule and the types of inputs. Therefore, some extensions from current research may be treated as future research works.

- The optimization of system decisions with respect to decision rule under different assumptions on failure rate of each voter is mentioned by Sah [25], and it turns out that the size of voters is strongly related. As the modern decision rule introduced in this chapter is much more

complicated, it requires more research on optimization study from the size of voters.

- The indecisive parameter is usually dynamic rather than static in an HO system; the true distributed indecisive parameter will maintain the system performance in a high level. In the meanwhile, the accurate distribution of indecisive parameter has significant influence on model optimization. More research is needed to address those problems.
- Inputs are assumed to be known and played an extremely important role in computation process in this chapter. However, this does not conclude all cases. In real, a WVS is required to generate the decision based on unknown input, and the problem of how to design the WVS given unknown inputs to keep the system function as reliable remains unsolved; a control system may be a key to be combined in WVS. Also, the state of inputs is another question to investigate, for example, the correlation between indecisive effect and continuous/discrete inputs remains unknown. Certain works need to be done for these meaningful problems.

References

1. Pham, H.: Reliability analysis of digital communication systems with imperfect voters. *Math. Comput. Model.* **26**(4), 103–112 (1997)
2. Gianaris, W.: Weighted voting in the International Monetary Fund and the World Bank. *Fordham International Law Journal.* **14**(4), 910–945 (1990)
3. Haghtalab, N., Noothigattu, R., Procaccia, A.D.: Weighted voting via no-regret learning. In: *Proceeding of 32nd Conference on Artificial Intelligence (AAAI)*, pp. 1055–1062 (2018)
4. Livieris, I., Kanavos, A., Tampakas, V., Pintelas, P.: A weighted voting ensemble self-labeled algorithm for the detection of lung abnormalities from X-rays. *Algorithms.* **12**(3), 64 (2019)
5. Levitin, G.: Analysis and optimization of weighted voting systems consisting of voting units with limited availability. *Reliab. Eng. Syst. Saf.* **73**(1), 91–100 (2001)
6. Pham, H., Malon, D.M.: Optimal design of systems with competing failure modes. *IEEE Trans. Reliab.* **43**, 251–254 (1994)
7. Pham, H.: Reliability analysis for dynamic configurations of systems with three failure modes. *Reliab. Eng. Syst. Saf.* **63**(1), 13–23 (1999)
8. Lucas, W.: Measuring power in weighted voting systems. In: *Political and Related Models Modules in Applied Mathematics*. Springer (1983)
9. Levitin, G.: Asymmetric weighted voting systems. *Reliab. Eng. Syst. Saf.* **76**(2), 205–212 (2002)
10. Aziz, H., Paterson, M., Leech, D.: Efficient algorithm for designing weighted voting games. In: *IEEE International Multitopic Conference*, pp. 1–6 (2007)
11. Nordmann, L., Pham, H.: Reliability of decision making in human-organizations. *IEEE Trans. Syst. Man Cybern. Syst. Hum.* **27**(4), 543–549 (1997)
12. Levitin, G.: Threshold optimization for weighted voting classifiers. *Nav. Res. Logist.* **50**(4), 322–344 (2003)
13. Posner, E., Sykes, A.: Voting rules in international organizations. *Chic. J. Int. Law.* **15**(1), 195–228 (2014)

14. Nordmann, L., Pham, H.: Weighted voting systems. *IEEE Trans. Reliab.* **48**(1), 42–49 (1999)
15. Xie, M., Pham, H.: Modeling the reliability of threshold weighted voting system. *Reliab. Eng. Syst. Saf.* **87**(1), 53–63 (2005)
16. Zhang, H., Pham, H.: Modeling reliability of threshold weighted indecisive voting systems. *IEEE Trans. Comput. Soc. Syst.* **7**(1), 35–41 (2020)
17. Levitin, G., Lisnianski, A.: Reliability optimization for weighted voting system. *Reliab. Eng. Syst. Saf.* **71**(2), 131–138 (2001)
18. Barnforff-Nielsen, O.E., Cox, D.R.: *Asymptotic Techniques for Use in Statistics*. Springer, Boston (1989)
19. Levitin, G.: Optimal unit grouping in weighted voting systems. *Reliab. Eng. Syst. Saf.* **72**(2), 179–191 (2001)
20. Long, Q., Xie, M., Ng, S.H., Levitin, G.: Reliability analysis and optimization of weighted voting systems with continuous states input. *Eur. J. Oper. Res.* **191**(1), 240–252 (2008)
21. Foroughi, A., Aouni, B.: New approaches for determining a common set of weights for a voting system. *Int. Trans. Oper. Res.* **19**(4), 495–520 (2012)
22. Sun, L., Han, C.Z.: Dynamic weighted voting for multiple classifier fusion: a generalized rough set method. *J. Syst. Eng. Electron.* **17**(3), 487–494 (2006)
23. Levitin, G.: Weighted voting systems: reliability versus rapidity. *Reliab. Eng. Syst. Saf.* **89**(2), 177–184 (2005)
24. Liu, Q., Zhang, H.: Weighted voting system with unreliable links. *IEEE Trans. Reliab.* **66**(2), 339–350 (2017)
25. Sah, R.: Fallibility in human organizations and political systems. *J. Econ. Perspect.* **5**(2), 67–88 (1991)



Hainan Zhang received degrees of BS and MS in Mathematics from Kansas State University in 2014, and MA degree in Statistics from Columbia University in 2015. He obtained PhD degree in the Department of Industrial and System Engineering. His current research interests include reliability modeling in weighted-voting systems and economic modeling in climate change scenarios.



Image Registration, Rigid Bodies, and Unknown Coordinate Systems

56

Ted Chang

Contents

56.1 Unknown Coordinate Systems and Their Estimation	1110
56.1.1 Problems of Unknown Coordinate Systems.....	1110
56.1.2 Image Registration.....	1111
56.1.3 The Orthogonal and Special Orthogonal Matrices.....	1111
56.1.4 The Procrustes and Spherical Regression Models.....	1112
56.1.5 Least Squares, L_1 , and M Estimation.....	1112
56.2 Least Squares Estimation	1113
56.2.1 Group Properties of $O(p)$ and $SO(p)$	1113
56.2.2 Singular Value Decomposition.....	1114
56.2.3 Least Squares Estimation in the Procrustes Model.....	1114
56.2.4 Example: Least Squares Estimates for the Hands Data.....	1115
56.2.5 Least Squares Estimation in the Spherical Regression Model.....	1115
56.3 Parameterizing $O(p)$ and $SO(p)$	1115
56.3.1 $SO(p)$ for $p = 2$	1116
56.3.2 $SO(p)$ for $p = 3$	1116
56.3.3 $SO(p)$ and $O(p)$, for General p , and the Matrix Exponential Map.....	1116
56.3.4 Distribution and Calculation of M -Estimates.....	1117
56.3.5 Numerical Calculation of M -Estimates for the Procrustes Model.....	1117
56.4 Statistical Properties of M-Estimates	1118
56.4.1 The Σ Matrix and the Geometry of the u_i	1118
56.4.2 Example: Σ for the Hands Data.....	1118
56.4.3 Statistical Assumptions for the Procrustes Model.....	1119
56.4.4 Theorem (Distribution of $(\hat{A}, \hat{\gamma}, \hat{b})$ for the Procrustes Model).....	1119
56.4.5 Example: A Test of $\gamma = 1$	1119
56.4.6 Example: A Test on A	1120
56.4.7 Asymptotic Relative Efficiency of Least Squares and L_1 Estimates.....	1121
56.4.8 The Geometry of the Landmarks and the Errors in \hat{A}	1121
56.4.9 Statistical Properties of M -Estimates for Spherical Regressions.....	1123
56.5 Diagnostics	1124
56.5.1 Influence Diagnostics in Simple Linear Regression.....	1124
56.5.2 Influence Diagnostics for the Procrustes Model.....	1125
56.5.3 Example: Influence for the Hands Data.....	1126
56.6 Location Problems and the Relationship to Regression Problems	1127
56.6.1 Location Problems Associated with Spherical and Procrustes Regressions.....	1127
56.6.2 Location and Regression Problems in Statistical Group Models.....	1127
References	1128

Abstract

This chapter deals with statistical problems involving *image registration* from landmark data, either in Euclidean 2-space R^2 or 3-space R^3 . In this problem, we have two images of the same object (such as satellite images taken at different times) or an image of a prototypical object and an actual object. It is desired to find the rotation, translation, and possibly scale change, which will best align the two images. Whereas many problems of this type are two-dimensional, it should be noted that medical imaging is often three dimensional.

After discussing several estimation techniques and their calculation, we discuss the relative efficiency of the various estimators. These results are important in choosing an optimal estimator. The relationship of the geometry of the landmarks to the statistical properties of the estimators is discussed. Finally we discuss diagnostics to determine which landmarks are most influential on the estimated registration. If the registration is unsatisfactory, these diagnostics can be used to determine which data points are most responsible and should be reexamined.

Keywords

Image registration · Influence function · Coordinate measuring machine · Multivariate normal distribution · Asymptotic relative efficiency

T. Chang (✉)
Department of Statistics, University of Virginia, Charlottesville, VA,
USA
e-mail: Tcc8v@virginia.edu

56.1 Unknown Coordinate Systems and Their Estimation

This chapter deals with statistical problems involving *image registration* from landmark data, either in Euclidean 2-space R^2 or 3-space R^3 . In this problem, we have two images of the same object (such as satellite images taken at different times) or an image of a prototypical object and an actual object. It is desired to find the rotation, translation, and possibly scale change, which will best align the two images. Whereas many problems of this type are two dimensional, it should be noted that medical imaging is often three-dimensional.

The same mathematics can be used to determine the errors in calculated motions of rigid bodies in Euclidean space, or in the determination of unknown Euclidean coordinates. The latter might occur, for example, if locations are measured relative to a moving object (“body coordinates”) and it is desired to reconcile these locations with respect to a fixed coordinate system (“space coordinates”).

This chapter focuses on image registrations. However, scientifically the most compelling example of rigid body motion is tectonic plates. To a first approximation, tectonic plates move as a rigid body on the unit sphere Ω_3 in R^3 . For this reason, we will also give results for Ω_3 and the unit circle Ω_2 . A separate chapter [8] in the present volume will focus primarily on tectonic plate reconstructions.

The chapter has six major sections.

After introducing some mathematical preliminaries, we introduce the concept of *M-estimators*, a generalization of least squares estimation. In least squares estimation, the registration that minimizes the sum of squares of the lengths of the deviations is chosen; in *M* estimation, the sum of squares of the lengths of the deviations is replaced by some other objective function. An important case is L_1 estimation, which minimizes the sum of the lengths of the deviations; L_1 estimation is often used when the possibility of outliers in the data is suspected.

The second section of this chapter deals with the calculation of least squares estimates. Then, in the third section, we introduce an iterative modification of the least squares algorithm to calculate other *M*-estimates. Note that minimization usually involves some form of differentiation and hence this section starts with a short introduction to the geometry of the group of rotations and differentiation in the rotation group. Many statistical techniques are based upon approximation by derivatives, and hence, a little understanding of geometry is necessary to understand the later statistical sections.

The fourth section discusses the statistical properties of *M*-estimates. A great deal of emphasis is placed upon the relationship between the geometric configuration of the landmarks and the statistical errors in the image registration. It is shown that these statistical errors are determined, up to a constant, by the geometry of the landmarks. The constant of

proportionality depends upon the objective function and the distribution of the errors in the data.

General statistical theory indicates that, if the data error distribution is (isotropic) multivariate normal, least squares estimation is optimal. An important result of this section is that, even in this case when least squares estimation is theoretically the most efficient, the use of L_1 estimation can guard against outliers with a very modest cost in efficiency. Here optimality and efficiency refer to the expected size of the statistical errors. In practice, data is often long-tailed and L_1 estimation yields *smaller* statistical errors than least squares estimation. This will be the case with the three-dimensional image registration example given here.

In the fifth section, we discuss diagnostics that can be used to determine which data points are most influential upon the registration. Thus, if the registration is unsatisfactory, these diagnostics can be used to determine which data points are most responsible and should be reexamined.

Finally, *Chang and Rivest* [1] discuss location and regression estimation in statistical group models and shows how some of the phenomenon discussed here can be shown to occur in a more general context. We discuss some of these results in a sixth section.

56.1.1 Problems of Unknown Coordinate Systems

Wahba [2] posed the following question. Suppose we have the directions of certain stars with respect to the unknown coordinate system of a satellite. How can we estimate the orientation of the satellite? Let \mathbf{A} be the unknown 3×3 matrix whose rows represent the axes of the satellite’s coordinate system with respect to a fixed and known (Earth) coordinate system. Furthermore, let \mathbf{u}_i be the directions of the stars with respect to the known coordinate systems, where each \mathbf{u}_i is written as a three-dimensional column vector with unit length. Similarly let \mathbf{v}_i be the directions of the stars with respect to the satellite’s coordinate system. Then

$$\mathbf{v}_i = \mathbf{A}\mathbf{u}_i + \text{error}. \quad (56.1)$$

In essence the question was to estimate \mathbf{A} . *Wahba* gave the least squares solution.

Chapman et al. [3] posed the same question in the following form. Suppose we have an object defined by a computer-aided design (CAD) program and a prototype is measured using a coordinate measuring machine (CMM). The orientations of lines on the object can be defined by unit vectors parallel to the lines and the orientations of planes can be defined by unit vectors normal to the planes. So we have unit vectors \mathbf{u}_i defined by the CAD program and the corresponding unit vectors \mathbf{v}_i as measured by the CMM. If \mathbf{A} is the coordinate

system of the CMM relative to the CAD program, then (56.1) holds.

Chapman et al. again used a least squares estimate $\hat{\mathbf{A}}$ of \mathbf{A} . The main question of interest, that is, the geometric integrity of the prototype, was then answered by analyzing the residuals of \mathbf{v}_i from $\hat{\mathbf{A}}\mathbf{u}_i$.

Since the \mathbf{u}_i and \mathbf{v}_i are of unit length, these two problems involve spherical data.

56.1.2 Image Registration

If we enlarge the inquiry to Euclidean space data, we arrive at the widely used *image registration* problem. Suppose $\mathbf{u}_i \in \mathbb{R}^p$ represent the locations of some landmarks in one image, and $\mathbf{v}_i \in \mathbb{R}^p$ the locations of corresponding landmarks in a second image of the same object. The usual applications occur with $p = 2, 3$. Under certain conditions, it might be reasonable to suppose that

$$\mathbf{v}_i = \mathbf{B}\mathbf{u}_i + \mathbf{b} + \text{error} \quad (56.2)$$

for an unknown $p \times p$ matrix \mathbf{B} and an unknown p -dimensional column vector \mathbf{b} . The matrix \mathbf{B} represents a coordinate change and the vector \mathbf{b} represents a translation of coordinates. The image registration problem is to estimate \mathbf{B} and \mathbf{b} .

The model (56.2) also arises in a slightly different context. Suppose we have landmarks \mathbf{u}_i on a prototypical face. For example, the \mathbf{u}_i might represent the locations of the nose, the two eyes, the base of the chin, etc. For the purpose of automated processing of a large number of facial images of different subjects, we might want to bring each facial image into alignment with the prototypical image using a transformation of the form (56.2) where the \mathbf{v}_i represents the same locations (nose, two eyes, base of chin, etc.) on the subject facial image.

In the absence of measurement error, one does not expect the landmarks on two faces to be related using a transformation of the form

$$\mathbf{v}_i = \mathbf{B}\mathbf{u}_i + \mathbf{b}. \quad (56.3)$$

The reader might be puzzled why a transformation of this form is under consideration. Statistical error, however, is not limited to measurement error. Statistical error incorporates all effects not included in the systematic portion of the model. In building a model of the form (56.2), we hope to separate out the most important relationship (56.3) between the landmarks \mathbf{u}_i on one object and the corresponding landmarks \mathbf{v}_i on the other object; the rest is placed in the statistical error.

Unlike the Wahba problem, the unknown (\mathbf{B}, \mathbf{b}) of the image registration problem, or the unknown \mathbf{A} in the *Chapman*

et al. problem, may not be of primary interest. Rather, they must be estimated as a preliminary step to more interesting problems. We will discuss herein the properties of various methods of estimating these unknowns. These properties will hopefully help the interested reader to choose a good estimation technique which will hopefully yield better results after this preliminary step is completed.

56.1.3 The Orthogonal and Special Orthogonal Matrices

Consider, for example, the dataset in Table 56.1 from *Chang* and *Ko* [4], which we will analyze repeatedly in what follows. This dataset consists of the digitized locations of 12 pairs of landmarks on the left and right hands of one of the authors. This is a $p = 3$ three-dimensional image registration problem. We might decide that, apart from the statistical error term, the shape of the two hands is the same; that is, the distance between two points on one hand is the same as the distance between the corresponding two points on the other hand.

This distance preserving condition translates mathematically to the equation $\mathbf{B}^T\mathbf{B} = \mathbf{I}_p$, the $p \times p$ -dimensional identity matrix. We outline a derivation of this well-known mathematical fact for the primary purpose of introducing the reader to the mathematical style of the remainder of this chapter. The distance between two p -dimensional column vectors \mathbf{v}_1 and \mathbf{v}_2 is

$$\|\mathbf{v}_2 - \mathbf{v}_1\| = \sqrt{(\mathbf{v}_2 - \mathbf{v}_1)^T(\mathbf{v}_2 - \mathbf{v}_1)}, \quad (56.4)$$

where the operations on the right-hand side of (56.4) are matrix multiplication and transposition. If the \mathbf{v}_i and \mathbf{u}_i are related by (56.3),

Table 56.1 12 digitized locations on the left and right hand

	Left hand u_i			Right hand v_i		
A	5.17	11.30	16.18	5.91	11.16	16.55
B	7.40	12.36	17.50	8.63	10.62	18.33
C	8.56	12.59	17.87	10.09	10.60	18.64
D	9.75	13.62	17.01	10.89	10.95	17.90
E	11.46	14.55	12.96	12.97	10.13	13.88
F	7.10	13.12	12.56	8.79	11.21	13.17
G	8.85	13.82	12.60	10.70	11.10	13.42
H	6.77	13.07	10.32	8.47	11.09	11.35
I	6.26	11.62	13.34	7.28	12.52	14.04
J	6.83	12.00	13.83	8.05	12.42	14.56
K	7.94	12.29	13.84	9.07	12.39	14.86
L	8.68	12.71	13.67	10.15	12.17	14.44

A top of little finger, *B* top of ring finger, *C* top of middle finger, *D* top of forefinger, *E* top of thumb, *F* gap between thumb and forefinger, *G* center of palm, *H* base of palm, *I* little finger knuckle, *J* ring finger knuckle, *K* middle finger knuckle, *L* forefinger knuckle

$$\begin{aligned} (\mathbf{v}_j - \mathbf{v}_i)^T (\mathbf{v}_j - \mathbf{v}_i) &= [\mathbf{B}(\mathbf{u}_j - \mathbf{u}_i)]^T [\mathbf{B}(\mathbf{u}_j - \mathbf{u}_i)] \\ &= (\mathbf{u}_j - \mathbf{u}_i)^T \mathbf{B}^T \mathbf{B} (\mathbf{u}_j - \mathbf{u}_i) \end{aligned}$$

Thus, if $\|\mathbf{v}_j - \mathbf{v}_i\| = \|\mathbf{u}_j - \mathbf{u}_i\|$ for all i and j , and if the \mathbf{u}_i do not all lie in a $(p - 1)$ -dimensional hyperplane of R^p ,

$$\mathbf{I}_p = \mathbf{B}^T \mathbf{B} = \mathbf{B} \mathbf{B}^T. \quad (56.5)$$

Note that the first equality of (56.5) implies that $\mathbf{B}^{-1} = \mathbf{B}^T$ and hence the second equality follows. Matrices which satisfy condition (56.5) are said to be *orthogonal*.

On the other hand, we might want to hypothesize that the two hands (again apart from statistical error) have the same shape except that one hand might be larger than the other. In this case, we are hypothesizing

$$\mathbf{B} = \gamma \mathbf{A} \quad (56.6)$$

where \mathbf{A} is orthogonal and γ is a positive real number.

In the Wahba and Chapman et al. problems, the rows of \mathbf{A} are known to be an orthonormal basis of R^3 . Since the (i, j) entry of $\mathbf{A} \mathbf{A}^T$ is the dot product of the i -th and j -th rows of \mathbf{A} , it follows that \mathbf{A} is orthogonal. However, more is known. Since the unknown coordinate system is known to be right-handed,

$$\mathbf{A}^T \mathbf{A} = \mathbf{I}_p, \quad \det(\mathbf{A}) = 1, \quad (56.7)$$

where $\det(\mathbf{A})$ is the determinant of the matrix \mathbf{A} . Such matrices are said to be *special orthogonal*.

In the hands data of Table 56.1, if we use the model (56.2) with condition (56.6), then \mathbf{A} will not be special orthogonal. This is because the left and right hands have different orientations. However, it is common in image registration problems to assume that condition (56.6) is true with \mathbf{A} assumed to be special orthogonal.

Following standard mathematical notation, we will use $O(p)$ to denote the $p \times p$ orthogonal matrices [i.e., the set of all matrices which satisfy (56.5)] and $SO(p)$ to denote the subset of $O(p)$ of special orthogonal matrices [i.e., the set of all matrices which satisfy (56.7)].

56.1.4 The Procrustes and Spherical Regression Models

In this chapter, we will be concerned with statistical methods which apply to the model (56.2) for Euclidean space data $\mathbf{u}_i, \mathbf{v}_i \in R^p$, for arbitrary p , where \mathbf{B} satisfies the condition (56.5) with \mathbf{B} constrained to be either orthogonal or special orthogonal. Following Goodall [5], we will call this model the *Procrustes* model.

We will also consider models of the form (56.1), where the p -vectors \mathbf{u}_i and \mathbf{v}_i are constrained to be of unit length, that is

$$\mathbf{u}_i, \mathbf{v}_i \in \Omega_p = S^{p-1} = \{\mathbf{x} \in R^p | \mathbf{x}^T \mathbf{x} = 1\}$$

and \mathbf{A} is constrained to be either orthogonal or special orthogonal. Following Chang [6], we will call this model the *spherical regression* model.

The statistical methodology for these two models can easily be described in parallel. In general, we will focus on the Procrustes model, while giving the modifications that apply to the spherical regression model.

56.1.5 Least Squares, L_1 , and M Estimation

In Sect. 56.2, we will derive the least squares estimate of \mathbf{A} , γ , \mathbf{b} for the Procrustes model. This estimate minimizes

$$\rho_2(\mathbf{A}, \gamma, \mathbf{b}) = \sum_i \|\mathbf{v}_i - \gamma \mathbf{A} \mathbf{u}_i - \mathbf{b}\|^2 \quad (56.8)$$

overall \mathbf{A} in either $O(p)$ or $SO(p)$, constants $\gamma > 0$, and p -vectors $\mathbf{b} \in R^p$. For the spherical regression model, the least squares estimate minimizes

$$\rho_2(\mathbf{A}) = \sum_i \|\mathbf{v}_i - \mathbf{A} \mathbf{u}_i\|^2 \quad (56.9)$$

$$= 2n - 2 \sum_i \mathbf{v}_i^T \mathbf{A} \mathbf{u}_i \quad (56.10)$$

overall \mathbf{A} in either $O(p)$ or $SO(p)$. For the second equality in (56.9), we have used that if $1 = \mathbf{v}^T \mathbf{v} = \mathbf{u}^T \mathbf{u}$, then

$$\begin{aligned} \|\mathbf{v} - \mathbf{A} \mathbf{u}\|^2 &= (\mathbf{v} - \mathbf{A} \mathbf{u})^T (\mathbf{v} - \mathbf{A} \mathbf{u}) \\ &= \mathbf{v}^T \mathbf{v} - \mathbf{v}^T \mathbf{A} \mathbf{u} - (\mathbf{A} \mathbf{u})^T \mathbf{v} + \mathbf{u}^T \mathbf{A}^T \mathbf{A} \mathbf{u} \\ &= 2 - 2\mathbf{v}^T \mathbf{A} \mathbf{u}. \end{aligned}$$

Least squares estimates have the advantage that an explicit closed-form solution for them is available. They have the disadvantage that they are very sensitive to *outliers*, that is, points $(\mathbf{u}_i, \mathbf{v}_i)$ for which the error term in (56.2) is unusually large. In the image registration problem, an outlier can arise in several contexts. It can be the result of a measurement error, or it can be the result of a misidentified landmark. Perhaps the image is not very clear, or the landmark (e.g., “point of the nose”) cannot be very precisely determined, or the landmark is obscured (by clouds or shrubs, etc.). Or perhaps there are places in the image where the image is not

really rigid, that is, the ideal match (56.3) does not apply very well. It is easy to conceive of a myriad of situations which might give rise to outliers.

L_1 estimators are often used to ameliorate the effects of outliers. These estimators minimize

$$\rho_1(\mathbf{A}, \gamma, \mathbf{b}) = \sum_i \|v_i - \gamma \mathbf{A}u_i - \mathbf{b}\|, \quad (56.11)$$

for the Procrustes model, or the sum of the distances along the surface of the sphere

$$\rho_1(\mathbf{A}) = \sum_i \arccos(v_i^T \mathbf{A}u_i) \quad (56.12)$$

for the spherical regression model. Unfortunately, an explicit closed-form solution for the L_1 estimate is not available and it must be calculated by numerical minimization. We will offer a few suggestions on approaches for numerical minimization in Sect. 56.3.5.

The least squares and L_1 estimators are special cases of the so-called M estimators. These estimators minimize an objective function of the form

$$\rho(\mathbf{A}, \gamma, \mathbf{b}) = \sum_i \rho_0(s_i), \quad (56.13)$$

where

$$s_i = \|v_i - \gamma \mathbf{A}u_i - \mathbf{b}\|$$

and ρ_0 is some increasing function. Intermediate between the least squares and L_1 estimate is the *Huber* estimate for which

$$\rho_0(s) = \begin{cases} (s/b)^2 & s < b \\ s/b & s \geq b \end{cases}$$

for some preset constant b . Or we can *Windsorize* the estimate

$$\rho_0(s) = \begin{cases} (s/b)^2 & s < b \\ 1 & s \geq b \end{cases}.$$

In the linear regression context, these and other objective functions are discussed in *Huber* [7].

For the spherical regression model, an M -estimator minimizes an objective function of the form

$$\rho(\mathbf{A}) = \sum_i \rho_0(t_i), \quad (56.14)$$

where

$$t_i = v_i^T \mathbf{A}u_i.$$

Notice that as v moves away from $\mathbf{A}u$ towards the antipodal point $-\mathbf{A}u$, $t = v^T \mathbf{A}u$ decreases from 1 to -1 . Thus, for the spherical case, $\rho_0(t)$ is chosen to be a decreasing function of t .

In Sect. 56.4 we will discuss the statistical properties of M -estimates. We will see how the geometry of the data translates into the error structure of the estimate. In the image registration problem, this information can be used, for example, to help select landmarks. General statistical theory indicates that under certain conditions (“normal distribution”) the least squares solution is optimal. However, if we were to use a L_1 estimate to guard against outliers, we would suffer a penalty of 13% for image registrations in two dimensions and only 8% for image registrations in three dimensions, even when least squares is theoretically optimal. We will make more precise in Sect. 56.4 how this *penalty* is defined. The important point to realize is that, especially for three-dimensional image registrations, L_1 estimators offer important protections against outliers in the data at very modest cost in the statistical efficiency of the estimator.

In Sect. 56.5, we will discuss diagnostics for the Procrustes and spherical regression models. If the image registration is not satisfactory, this section will give tools to determine which of the landmarks is causing the unsatisfactory registration. It will follow, for example, that landmarks that greatly influence \mathbf{A} will have negligible influence on γ and vice versa.

56.2 Least Squares Estimation

56.2.1 Group Properties of $O(p)$ and $SO(p)$

It is important to note that $O(p)$ and $SO(p)$ are groups in the mathematical sense. That is, if $\mathbf{A}, \mathbf{B} \in O(p)$, then

$$(\mathbf{AB})^T (\mathbf{AB}) = \mathbf{B}^T \mathbf{A}^T \mathbf{A} \mathbf{B} = \mathbf{B}^T \mathbf{I}_p \mathbf{B} = \mathbf{I}_p$$

since both \mathbf{A} and \mathbf{B} satisfy (56.5). Thus, $\mathbf{AB} \in O(p)$. Similarly if $\mathbf{A} \in O(p)$, then (56.5) implies that $\mathbf{A}^{-1} = \mathbf{A}^T \in O(p)$. This implies that $O(p)$ is a group. Furthermore, if $\det(\mathbf{A}) = \det(\mathbf{B}) = 1$, then $\det(\mathbf{AB}) = \det(\mathbf{A})\det(\mathbf{B}) = 1$ and $\det(\mathbf{A}^{-1}) = 1/\det(\mathbf{A}) = 1$. In summary, we have

$$\begin{aligned} \text{If } \mathbf{A}, \mathbf{B} \in O(p), \quad \text{then } \mathbf{AB} \in O(p) \\ \text{and } \mathbf{A}^{-1} = \mathbf{A}^T \in O(p) \\ \text{If } \mathbf{A}, \mathbf{B} \in SO(p), \quad \text{then } \mathbf{AB} \in SO(p) \\ \text{and } \mathbf{A}^{-1} = \mathbf{A}^T \in SO(p). \end{aligned} \quad (56.15)$$

Notice also that, if \mathbf{A} satisfies (56.5), then $1 = \det(\mathbf{A}^T \mathbf{A}) = [\det(\mathbf{A})]^2$ so that $\det(\mathbf{A}) = 1, -1$.

56.2.2 Singular Value Decomposition

Given a $p \times q$ matrix \mathbf{X} its *singular value decomposition* is

$$\mathbf{X} = \mathbf{O}_1 \mathbf{\Lambda} \mathbf{O}_2^T \quad (56.16)$$

where $\mathbf{O}_1 \in O(p)$, $\mathbf{O}_2 \in O(q)$ and $\mathbf{\Lambda}$ is $p \times q$. If $p \leq q$, $\mathbf{\Lambda}$ has block form

$$\mathbf{\Lambda} = \begin{bmatrix} \text{diag}(\lambda_1, \dots, \lambda_p) & \mathbf{0}_{(p, q-p)} \end{bmatrix}$$

Here $\text{diag}(\lambda_1, \dots, \lambda_p)$ is a diagonal matrix with entries $\lambda_1 \geq \dots \geq \lambda_p$ and $\mathbf{0}_{(p, q-p)}$ is a $p \times (q - p)$ matrix with all zeros. If $q \leq p$

$$\mathbf{\Lambda} = \begin{pmatrix} \text{diag}(\lambda_1, \dots, \lambda_q) \\ \mathbf{0}_{(p-q, q)} \end{pmatrix}.$$

Most mathematical software packages now include the singular value decomposition. However, it can be computed using a package which only computes eigen-decompositions of symmetric matrices. Suppose temporarily $p \leq q$. Since $\mathbf{X}\mathbf{X}^T$ is a symmetric nonnegative definite matrix, its eigen-decomposition has the form

$$\mathbf{X}\mathbf{X}^T = \mathbf{O}_1 \mathbf{\Lambda}_1 \mathbf{O}_1^T$$

where $\mathbf{O}_1 \in O(p)$ and $\mathbf{\Lambda}_1 = \text{diag}(\lambda_1^2, \dots, \lambda_p^2)$ with $\lambda_1 \geq \dots \geq \lambda_p \geq 0$. The columns of \mathbf{O}_1 are the eigenvectors of $\mathbf{X}\mathbf{X}^T$ and $\lambda_1^2, \dots, \lambda_p^2$ are the corresponding eigenvalues. Suppose $\lambda_p > 0$ and let $\tilde{\mathbf{O}}_2 = \mathbf{X}^T \mathbf{O}_1 \mathbf{\Lambda}_1^{-1/2}$. $\tilde{\mathbf{O}}_2$ is $q \times p$, but

$$\begin{aligned} \tilde{\mathbf{O}}_2^T \tilde{\mathbf{O}}_2 &= \mathbf{\Lambda}_1^{-1/2} \mathbf{O}_1^T \mathbf{X}\mathbf{X}^T \mathbf{O}_1 \mathbf{\Lambda}_1^{-1/2} \\ &= \mathbf{\Lambda}_1^{-1/2} \mathbf{O}_1^T \mathbf{O}_1 \mathbf{\Lambda}_1 \mathbf{O}_1^T \mathbf{O}_1 \mathbf{\Lambda}_1^{-1/2} \\ &= \mathbf{\Lambda}_1^{-1/2} \mathbf{\Lambda}_1 \mathbf{\Lambda}_1^{-1/2} = \mathbf{I}_p, \end{aligned}$$

so that the columns of $\tilde{\mathbf{O}}_2$ are orthonormal. Furthermore

$$\mathbf{O}_1 \mathbf{\Lambda}_1^{1/2} \tilde{\mathbf{O}}_2^T = \mathbf{O}_1 \mathbf{\Lambda}_1^{1/2} \mathbf{\Lambda}_1^{-1/2} \mathbf{O}_1^T \mathbf{X} = \mathbf{X}.$$

Filling $\mathbf{\Lambda}_1^{1/2}$ with $q - p$ columns of zeros, and completing the columns of $\tilde{\mathbf{O}}_2$ to an orthonormal basis of R^q yields the decomposition (56.16).

Extensions to the cases when $\lambda_p = 0$ or when $q \leq p$ will not be difficult for the careful reader.

56.2.3 Least Squares Estimation in the Procrustes Model

The least squares estimation of the Procrustes model (56.2) has long been known (see, for example, Goodall [5]). Let

$\bar{\mathbf{u}} = n^{-1} \sum_i \mathbf{u}_i$, where n is the number of pairs $(\mathbf{u}_i, \mathbf{v}_i)$ and let $\bar{\mathbf{v}}$ be similarly defined. Define the $p \times p$ matrix \mathbf{X} by

$$\mathbf{X} = \sum_i (\mathbf{u}_i - \bar{\mathbf{u}}) (\mathbf{v}_i - \bar{\mathbf{v}})^T.$$

Then

$$\begin{aligned} \rho_2(\mathbf{A}, \gamma, \mathbf{b}) &= \sum_i \|\mathbf{v}_i - \gamma \mathbf{A} \mathbf{u}_i - \mathbf{b}\|^2 \\ &= \sum_i \left\| \mathbf{v}_i - \bar{\mathbf{v}} - \gamma \mathbf{A} (\mathbf{u}_i - \bar{\mathbf{u}}) - [\mathbf{b} - (\bar{\mathbf{v}} - \gamma \mathbf{A} \bar{\mathbf{u}})] \right\|^2 \\ &= \sum_i \|\mathbf{v}_i - \bar{\mathbf{v}}\|^2 \\ &\quad - \gamma \sum_i (\mathbf{u}_i - \bar{\mathbf{u}})^T \mathbf{A}^T (\mathbf{v}_i - \bar{\mathbf{v}}) \\ &\quad - \gamma \sum_i (\mathbf{v}_i - \bar{\mathbf{v}})^T \mathbf{A} (\mathbf{u}_i - \bar{\mathbf{u}}) \\ &\quad + \gamma^2 \sum_i \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2 \\ &\quad + n \|\mathbf{b} - (\bar{\mathbf{v}} - \gamma \mathbf{A} \bar{\mathbf{u}})\|^2. \end{aligned}$$

All the other cross-product terms sum to zero. Now

$$\begin{aligned} \sum_i (\mathbf{v}_i - \bar{\mathbf{v}})^T \mathbf{A} (\mathbf{u}_i - \bar{\mathbf{u}}) &= \sum_i \text{Tr} [(\mathbf{v}_i - \bar{\mathbf{v}})^T \mathbf{A} (\mathbf{u}_i - \bar{\mathbf{u}})] \\ &= \sum_i \text{Tr} [\mathbf{A} (\mathbf{u}_i - \bar{\mathbf{u}}) (\mathbf{v}_i - \bar{\mathbf{v}})^T] = \text{Tr}(\mathbf{A}\mathbf{X}) \end{aligned}$$

and

$$\begin{aligned} \sum_i (\mathbf{u}_i - \bar{\mathbf{u}})^T \mathbf{A}^T (\mathbf{v}_i - \bar{\mathbf{v}}) &= \sum_i (\mathbf{v}_i - \bar{\mathbf{v}})^T \mathbf{A} (\mathbf{u}_i - \bar{\mathbf{u}}) = \text{Tr}(\mathbf{A}\mathbf{X}). \end{aligned}$$

Therefore

$$\begin{aligned} \rho_2(\mathbf{A}, \gamma, \mathbf{b}) &= \sum_i \|\mathbf{v}_i - \bar{\mathbf{v}}\|^2 - 2\gamma \text{Tr}(\mathbf{A}\mathbf{X}) \\ &\quad + \gamma^2 \sum_i \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2 \\ &\quad + n \|\mathbf{b} - (\bar{\mathbf{v}} - \gamma \mathbf{A} \bar{\mathbf{u}})\|^2. \end{aligned} \quad (56.17)$$

Substituting (56.16),

$$\begin{aligned} \text{Tr}(\mathbf{A}\mathbf{X}) &= \text{Tr}(\mathbf{A} \mathbf{O}_1 \mathbf{\Lambda} \mathbf{O}_2^T) = \text{Tr}(\mathbf{O}_2^T \mathbf{A} \mathbf{O}_1 \mathbf{\Lambda}) \\ &= \sum_i \lambda_i e_{ii}, \end{aligned}$$

where e_{ii} are the diagonal entries of $\mathbf{O}_2^T \mathbf{A} \mathbf{O}_1 \in O(p)$. Now $|e_{ii}| \leq 1$ and hence $\text{Tr}(\mathbf{A}\mathbf{X})$ is maximized when $e_{ii} = 1$ or, equivalently, when $\mathbf{O}_2^T \mathbf{A} \mathbf{O}_1 = \mathbf{I}_p$. This implies $\mathbf{A} = \mathbf{O}_2 \mathbf{O}_1^T$.

Thus, if $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$ minimizes (56.17),

$$\begin{aligned}\hat{\mathbf{A}} &= \mathbf{O}_2 \mathbf{O}_1^T, \\ \hat{\gamma} &= (\sum_i \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2)^{-1} \text{Tr}(\hat{\mathbf{A}}\mathbf{X}) \\ &= (\sum_i \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2)^{-1} \sum_i \lambda_i, \\ \hat{\mathbf{b}} &= \bar{\mathbf{v}} - \hat{\gamma} \hat{\mathbf{A}} \bar{\mathbf{u}}.\end{aligned}\quad (56.18)$$

If \mathbf{A} is constrained to lie in $SO(p)$, we use a modified singular value decomposition. Let $\mathbf{X} = \tilde{\mathbf{O}}_1 \tilde{\mathbf{\Lambda}} \tilde{\mathbf{O}}_2$ be the (usual) singular value decomposition of \mathbf{X} and let

$$\mathbf{E} = \text{diag}(1, \dots, 1, -1) \quad (56.19)$$

be the identity matrix with its last entry changed to -1 . Let $\mathbf{O}_1 = \tilde{\mathbf{O}}_1 \mathbf{E}^{\delta_1}$ where $\delta_1 = 0$ if $\tilde{\mathbf{O}}_1 \in SO(p)$ and $\delta_1 = 1$ otherwise.

Similarly define δ_2 and \mathbf{O}_2 . Finally write $\mathbf{A} = \tilde{\mathbf{\Lambda}} \mathbf{E}^{\delta_1 + \delta_2}$. Then (56.16) is valid with $\mathbf{O}_1, \mathbf{O}_2 \in SO(p)$ and $\lambda_1 \geq \dots \geq \lambda_{p-1} \geq |\lambda_p|$.

This is the modified singular value decomposition.

The least squares estimates, subject to the constraint $\hat{\mathbf{A}} \in SO(p)$, is still given by (56.18) when a modified singular value decomposition is used for \mathbf{X} .

56.2.4 Example: Least Squares Estimates for the Hands Data

Consider, for example, the hands data in Table 56.1. For this data

$$\begin{aligned}\bar{\mathbf{u}} &= \begin{bmatrix} 7.8975 \\ 12.7542 \\ 14.3067 \end{bmatrix}, \quad \bar{\mathbf{v}} = \begin{bmatrix} 9.2500 \\ 11.3633 \\ 15.0950 \end{bmatrix}, \\ \mathbf{X} &= \left(\begin{bmatrix} 5.17 \\ 11.30 \\ 16.18 \end{bmatrix} - \bar{\mathbf{u}} \right) \left(\begin{bmatrix} 5.91 \\ 11.16 \\ 16.55 \end{bmatrix} - \bar{\mathbf{v}} \right)^T + \dots + \\ &\quad \left(\begin{bmatrix} 8.68 \\ 12.71 \\ 13.67 \end{bmatrix} - \bar{\mathbf{u}} \right) \left(\begin{bmatrix} 10.15 \\ 12.17 \\ 14.44 \end{bmatrix} - \bar{\mathbf{v}} \right)^T \\ &= \begin{bmatrix} 34.0963 & -6.9083 & 3.5769 \\ 17.3778 & -4.9028 & -5.6605 \\ -2.3940 & -5.7387 & 57.8598 \end{bmatrix}.\end{aligned}$$

The singular value decomposition $\mathbf{X} = \mathbf{O}_1 \mathbf{\Lambda} \mathbf{O}_2^T$ is given by

$$\mathbf{O}_1 = \begin{bmatrix} 0.0465 & -0.8896 & -0.4544 \\ -0.1012 & -0.4567 & 0.8838 \\ 0.9938 & -0.0048 & 0.1112 \end{bmatrix}$$

$$\mathbf{O}_2 = \begin{bmatrix} -0.0436 & -0.9764 & -0.2114 \\ -0.0944 & 0.2147 & -0.9721 \\ 0.9946 & -0.0224 & -0.1015 \end{bmatrix}$$

$$\mathbf{\Lambda} = \text{diag}(58.5564 \quad 39.1810 \quad 1.8855)$$

Hence (56.18) yields

$$\hat{\mathbf{A}} = \begin{bmatrix} 0.9627 & 0.2635 & -0.0621 \\ 0.2463 & -0.9477 & -0.2030 \\ 0.1123 & -0.1801 & 0.9722 \end{bmatrix}$$

$$\hat{\gamma} = 0.9925$$

$$\hat{\mathbf{b}} = [-0.7488 \quad 24.3115 \quad 2.6196]^T. \quad (56.20)$$

Notice that $\det(\hat{\mathbf{A}}) = -1$, so $\hat{\mathbf{A}} \notin SO(3)$. We expect this result since, as previously remarked, the left and right hands have different orientations. The value of $\hat{\gamma}$ is somewhat puzzling since the subject is right-handed and one would expect; therefore, $\gamma > 1$. Although, as we will see in Sect. 56.4, the difference between $\hat{\gamma}$ and 1 is not significant, a better estimate would have been achieved if the L_1 objective function (56.11) were numerically minimized instead. In this case $\hat{\gamma} = 1.0086$. Our analysis will show that the hands dataset has an outlier and we see here an example of the superior resistance of L_1 estimates to outliers.

56.2.5 Least Squares Estimation in the Spherical Regression Model

Least squares estimation for the spherical regression model is similar to least squares estimation in the Procrustes model. Let $\mathbf{X} = \sum_i \mathbf{u}_i \mathbf{v}_i^T$ and define $\mathbf{O}_1, \mathbf{O}_2 \in O(p)$ using a singular value decomposition of \mathbf{X} . Then $\hat{\mathbf{A}} = \mathbf{O}_2 \mathbf{O}_1^T$. If, on the other hand, it is desired to constrain $\hat{\mathbf{A}}$ to $SO(p)$, one defines $\mathbf{O}_1, \mathbf{O}_2 \in SO(p)$ using a modified singular value decomposition and, again, $\hat{\mathbf{A}} = \mathbf{O}_2 \mathbf{O}_1^T$.

56.3 Parameterizing $O(p)$ and $SO(p)$

To develop the statistical properties of $\hat{\mathbf{A}}$, it is necessary to rewrite $O(p)$ and $SO(p)$ into a smaller number of parameters.

56.3.1 $SO(p)$ for $p = 2$

For $p = 2$,

$$SO(2) = \left\{ \Phi_2(h) = \begin{pmatrix} \cos(h) & -\sin(h) \\ \sin(h) & \cos(h) \end{pmatrix} \mid h \in \mathbb{R}^1 \right\}. \quad (56.21)$$

Physically $\Phi_2(h)$ represents a rotation of \mathbb{R}^2 by an angle of h radians. Since $\Phi_2(h) = \Phi_2(h + 2\pi)$, $SO(2)$ is geometrically a circle.

Since each element of $SO(2)$ has four entries, it is tempting to think of $SO(2)$ as four-dimensional. However, as (56.21) makes clear, $SO(2)$ can be described by one parameter $h \in \mathbb{R}^1$. Thus, $SO(2)$ is really one-dimensional. Suppose we were constrained to live on a circle Ω_2 (instead of the sphere Ω_3). At each point on Ω_2 we can only travel to our left or to our right, and, if our travels were limited, it would appear as if we only had one-dimensional travel. Mathematicians describe this situation by saying that $SO(2)$ is a one-dimensional *manifold*. Manifolds, and their importance to geometric statistical problems, are further discussed in the accompanying chapter [8].

Notice also $\Phi_2(0) = \mathbf{I}_2$ and that, if h is small, then $\Phi_2(h)$ is close to \mathbf{I}_2 . Thus, if h is small, $\Phi_2(h)\mathbf{x}$ is close to \mathbf{x} for all $\mathbf{x} \in \mathbb{R}^2$. As we shall see, this simple observation is key to understanding our approach to the statistical properties of $\hat{\mathbf{A}}$.

56.3.2 $SO(p)$ for $p = 3$

$SO(3)$ can be described as the collection of all rotations in \mathbb{R}^3 . That is,

$$SO(3) = \{ \Phi_3(\mathbf{h}) \mid \mathbf{h} \in \mathbb{R}^3 \} \quad (56.22)$$

where $\Phi_3(\mathbf{h})$ is right-hand rule rotation of $\|\mathbf{h}\|$ radians around the axis $\|\mathbf{h}\|^{-1}\mathbf{h}$. Writing $\theta = \|\mathbf{h}\|$ and $\boldsymbol{\xi} = \|\mathbf{h}\|^{-1}\mathbf{h}$, so that $\boldsymbol{\xi}$ is a unit-length three-dimensional vector, it can be shown that

$$\begin{aligned} \Phi_3(\mathbf{h}) = \Phi_3(\theta\boldsymbol{\xi}) &= \cos(\theta) \mathbf{I}_3 + \sin(\theta) M_3(\boldsymbol{\xi}) \\ &+ (1 - \cos(\theta)) \boldsymbol{\xi}\boldsymbol{\xi}^T \end{aligned} \quad (56.23)$$

Where

$$M_3(\boldsymbol{\xi}) = M_3 \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix} = \begin{pmatrix} 0 & -\xi_3 & \xi_2 \\ \xi_3 & 0 & -\xi_1 \\ -\xi_2 & \xi_1 & 0 \end{pmatrix}.$$

Thus, although each $\mathbf{A} \in SO(3)$ has nine entries, $SO(3)$ is actually a three-dimensional manifold.

Again we notice that $\Phi_3(\mathbf{0}) = \mathbf{I}_3$ and that if $\|\mathbf{h}\|$ is small then $\Phi_3(\mathbf{h})\mathbf{x}$ is close to \mathbf{x} for all $\mathbf{x} \in \mathbb{R}^3$.

For future use, we note that if $\mathbf{C} \in SO(3)$, then the axis $\boldsymbol{\xi}$ of the rotation represented by \mathbf{C} satisfies $\mathbf{C}\boldsymbol{\xi} = \boldsymbol{\xi}$. Thus, $\boldsymbol{\xi}$ is the eigenvector associated to the eigenvalue 1 of \mathbf{C} . By re-representing \mathbf{C} in an orthonormal basis which includes $\boldsymbol{\xi}$, one can show that the angle of rotation θ of the rotation represented by \mathbf{C} satisfies $1 + 2\cos(\theta) = \text{Tr}(\mathbf{C})$. Thus, if $\boldsymbol{\xi}$ and θ are calculated in this way, $\Phi_3(\theta\boldsymbol{\xi}) = \mathbf{C}$.

56.3.3 $SO(p)$ and $O(p)$, for General p , and the Matrix Exponential Map

For general p , let \mathbf{H} be a $p \times p$ skew-symmetric matrix; that is

$$\mathbf{H}^T = -\mathbf{H}.$$

We define the matrix exponential map by

$$\exp(\mathbf{H}) = \sum_{k=0}^{k=\infty} \frac{\mathbf{H}^k}{k!}.$$

It can be shown that the skew-symmetry condition implies that $\exp(\mathbf{H})[\exp(\mathbf{H})]^T = \mathbf{I}_p$ and indeed

$$SO(p) = \{ \exp(\mathbf{H}) \mid \mathbf{H} \text{ is skew symmetric} \} \quad (56.24)$$

A skew-symmetric matrix must have zeros on its main diagonal and its entries below the main diagonal are determined by its entries above the main diagonal. Thus, the skew-symmetric $p \times p$ matrices have $p(p-1)/2$ independent entries, and hence, $SO(p)$ is a manifold of dimension $p(p-1)/2$.

Let $\mathbf{0}$ be a $p \times p$ matrix of zeros. Then

$$\exp(\mathbf{0}) = \mathbf{I}_p. \quad (56.25)$$

Thus, if the entries of \mathbf{H} are small (in absolute value), then $\exp(\mathbf{H})$ will be close to the identity matrix.

For $p = 3$, it can be shown, by using (56.23), that $\Phi_3(\mathbf{h}) = \exp[M_3(\mathbf{h})]$ for $\mathbf{h} \in \mathbb{R}^3$. Similarly we define for $h \in \mathbb{R}^1$ the skew-symmetric matrix

$$M_2(h) = \begin{pmatrix} 0 & -h \\ h & 0 \end{pmatrix}$$

and it follows that $\Phi_2(h) = \exp[M_2(h)]$. Thus, (56.21) and (56.22) are indeed special cases of (56.24).

$O(p)$ has two connected components; one is $SO(p)$ and the other is

$$SO(p)E = \{ \mathbf{A}\mathbf{E} \mid \mathbf{A} \in SO(p) \},$$

where E has been previously defined in (56.19). Notice that E is a reflection of R^p through the $(p - 1)$ -dimensional hyperplane perpendicular to the last coordinate vector. Indeed all reflections of R^p are in $O(p)$.

56.3.4 Distribution and Calculation of M -Estimates

So, heuristically speaking, suppose we have estimates $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$ which minimize an objective function of the form (56.13). What values of the unknown parameters $(\mathbf{A}, \gamma, \mathbf{b})$ should we consider as reasonable given the data? The obvious answer, which is fully consistent with the usual practices of statistics, is those $(\mathbf{A}, \gamma, \mathbf{b})$ which do not excessively degrade the fit of the best-fit parameters $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$; that is, those $(\mathbf{A}, \gamma, \mathbf{b})$ for which

$$\rho(\mathbf{A}, \gamma, \mathbf{b}) - \rho(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}}) = \sum_i \left[\rho_0(\|v_i - \gamma \mathbf{A} u_i - \mathbf{b}\|) - \rho_0(\|v_i - \hat{\gamma} \hat{\mathbf{A}} u_i - \hat{\mathbf{b}}\|) \right]$$

is not too large.

Recall that, for $p = 3$, if \mathbf{h} is small, then $\Phi_3(\mathbf{h})\mathbf{u}_i$ will be close to \mathbf{u}_i . This suggests writing

$$\hat{\mathbf{A}} = \mathbf{A} \Phi_3(\hat{\mathbf{h}}), \quad (56.26)$$

where $\hat{\mathbf{h}} \in R^3$. Then $\mathbf{A} \mathbf{u}_i = \hat{\mathbf{A}} \Phi_3(-\hat{\mathbf{h}})\mathbf{u}_i$ will be close to $\hat{\mathbf{A}} \mathbf{u}_i$ when $\hat{\mathbf{h}}$ is small. Rather than focus on the distribution of $\hat{\mathbf{A}}$, we will focus on the distribution of the *deviation* of $\hat{\mathbf{A}}$ from \mathbf{A} as measured by the (hopefully) small vector $\hat{\mathbf{h}}$.

Similarly, for $p = 2$, we will write

$$\hat{\mathbf{A}} = \mathbf{A} \Phi_2(\hat{\mathbf{h}}), \quad (56.27)$$

where $\hat{\mathbf{h}} \in R^1$. For general p , one writes

$$\hat{\mathbf{A}} = \mathbf{A} \exp(\hat{\mathbf{H}}), \quad (56.28)$$

where $\hat{\mathbf{H}}$ is $p \times p$ skew-symmetric.

The most elementary procedures in statistics are based upon the fact

If X_1, \dots, X_n are independent and each X_i is distributed $N(\mu, \sigma^2)$, then \bar{X} is distributed $N(\mu, \sigma^2/n)$.

An equivalent result is

If X_1, \dots, X_n are independent and each X_i is distributed $N(\mu, \sigma^2)$, then $\bar{X} - \mu$ is distributed $N(0, \sigma^2/n)$.

In the latter form, we have an estimator (in this case \bar{X}) and the distribution of the *deviation* $\hat{h} = \bar{X} - \mu$ of the estimator from the unknown parameter μ . This is sufficient for both confidence intervals and hypothesis testing and is analogous to what we propose to do in Sect. 56.4.

Similarly, in what follows, we will focus on the distributions of $\hat{h}, \hat{\mathbf{h}}$ or $\hat{\mathbf{H}}$ (depending upon whether $p = 2, 3$, or general) instead of directly giving the distributions of $\hat{\mathbf{A}}$. The reasons for doing so are rooted in the geometry of $SO(p)$ and are explained in greater detail in Sects. 53.3 and 53.4 of the accompanying chapter [8].

56.3.5 Numerical Calculation of M -Estimates for the Procrustes Model

We use here the geometric insights into $SO(p)$ to propose a method of minimizing the objective function (56.13) for the Procrustes model. The simplifications necessary to minimize the objective function (56.14) for the spherical regression model should be reasonably clear.

In what follows, it will be convenient to rewrite the Procrustes model

$$v_i = \gamma \mathbf{A} u_i + \mathbf{b} + \text{error}$$

in the equivalent form

$$v_i = \gamma \mathbf{A} (u_i - \bar{u}) + \beta + \text{error}, \quad (56.29)$$

where $\beta = \gamma \mathbf{A} \bar{u} + \mathbf{b}$.

Let $\psi(s) = \rho'_0(s)$. Differentiating (56.13) with respect to γ and β , we get that the M -estimates $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\beta})$ must satisfy

$$0 = \sum_i \psi(s_i) s_i^{-1} \left[v_i - \hat{\gamma} \hat{\mathbf{A}} (u_i - \bar{u}) - \hat{\beta} \right]^T \hat{\mathbf{A}} (u_i - \bar{u}) \quad (56.30)$$

$$0 = \sum_i \psi(s_i) s_i^{-1} \left[v_i - \hat{\gamma} \hat{\mathbf{A}} (u_i - \bar{u}) - \hat{\beta} \right]^T \quad (56.31)$$

where $s_i = \|v_i - \hat{\gamma} \hat{\mathbf{A}} (u_i - \bar{u}) - \hat{\beta}\|$.

To differentiate (56.13) with respect to \mathbf{A} , we note that, if \mathbf{H} is any skew-symmetric matrix, and using (56.25),

$$\begin{aligned} 0 &= \frac{d}{dt} \Big|_{t=0} \left\{ \sum_i \rho_0 \left(\|v_i - \hat{\gamma} \hat{\mathbf{A}} \exp(t\mathbf{H})(u_i - \bar{u}) - \hat{\beta}\| \right) \right\} \\ &= -\hat{\gamma} \sum_i \psi(s_i) s_i^{-1} \left[v_i - \hat{\gamma} \hat{\mathbf{A}} (u_i - \bar{u}) - \hat{\beta} \right]^T \hat{\mathbf{A}} \mathbf{H} (u_i - \bar{u}) \\ &= -\hat{\gamma} \text{Tr}(\tilde{\mathbf{X}} \hat{\mathbf{A}} \mathbf{H}), \end{aligned}$$

where

$$\tilde{\mathbf{X}} = \sum_i \psi(s_i) s_i^{-1} (\mathbf{u}_i - \bar{\mathbf{u}}) \left[\mathbf{v}_i - \hat{\gamma} \hat{\mathbf{A}} (\mathbf{u}_i - \bar{\mathbf{u}}) - \hat{\boldsymbol{\beta}} \right]^T.$$

Since \mathbf{H} is any skew-symmetric matrix, $\tilde{\mathbf{X}}\hat{\mathbf{A}}$ is symmetric. Equivalently if

$$\mathbf{X} = \sum_i \psi(s_i) s_i^{-1} (\mathbf{u}_i - \bar{\mathbf{u}}) (\mathbf{v}_i - \hat{\boldsymbol{\beta}})^T \quad (56.32)$$

then $\mathbf{X}\hat{\mathbf{A}}$ is symmetric.

Equations (56.32), (56.30), and (56.31) lead to the following iterative minimization algorithm. Start with the least squares solution given in Sect. 56.2.3 and use these estimates to calculate s_i . Using these s_i and the current guess for $\hat{\mathbf{A}}$, solve (56.30) and (56.31) to update the guesses for $\hat{\gamma}$ and $\hat{\boldsymbol{\beta}}$. Now writing $\mathbf{X} = \mathbf{O}_1 \mathbf{\Lambda} \mathbf{O}_2^T$ for the singular value decomposition of \mathbf{X} , the next guess for $\hat{\mathbf{A}}$ is $\mathbf{O}_2 \mathbf{O}_1^T$. This yields a minimum in $O(p)$. If minimization in $\text{SO}(p)$ is desired, a modified singular value decomposition is used for \mathbf{X} instead. Having updated the guesses for $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\boldsymbol{\beta}})$, we now iterate.

For example, consider the hands data of Table 56.1. We calculate the L_1 estimate for which $\psi(s) = 1$. Starting with the least squares estimates in (56.20), we convert $\hat{\mathbf{b}}$ to

$$\hat{\boldsymbol{\beta}} = \hat{\gamma} \hat{\mathbf{A}} \bar{\mathbf{u}} + \hat{\mathbf{b}} = (9.2500 \ 11.3633 \ 15.0950)^T \quad (56.33)$$

We use these least squares estimate as an initial guess; a single iteration of the minimization algorithm yields the updated guess

$$\hat{\mathbf{A}} = \begin{pmatrix} 0.9569 & 0.2823 & -0.0690 \\ 0.2614 & -0.9399 & -0.2199 \\ 0.1269 & -0.1924 & 0.9731 \end{pmatrix},$$

$$\hat{\gamma} = 1.0015,$$

$$\hat{\boldsymbol{\beta}} = (9.2835 \ 11.4092 \ 15.0851)^T.$$

Convergence is achieved after around a dozen iterations. We arrive at the L_1 estimates

$$\hat{\mathbf{A}} = \begin{pmatrix} 0.9418 & 0.3274 & -0.0760 \\ 0.3045 & -0.9268 & -0.2200 \\ 0.1425 & -0.1840 & 0.9725 \end{pmatrix},$$

$$\hat{\gamma} = 1.0086, \quad (56.34)$$

$$\hat{\boldsymbol{\beta}} = (9.2850 \ 11.4255 \ 15.0883)^T.$$

56.4 Statistical Properties of M -Estimates

56.4.1 The Σ Matrix and the Geometry of the \mathbf{u}_i

Let Σ be the $p \times p$ matrix

$$\Sigma = n^{-1} \sum_i (\mathbf{u}_i - \bar{\mathbf{u}}) (\mathbf{u}_i - \bar{\mathbf{u}})^T$$

Σ is nonnegative definite symmetric and hence its eigenvalues are real and its eigenvectors form an orthonormal basis of R^p . We can use this eigen-decomposition of Σ to summarize the geometry of the point \mathbf{u}_i . More specifically, let $\lambda_1 \geq \dots \geq \lambda_p \geq 0$ be the eigenvalues of Σ with corresponding eigenvectors $\mathbf{e}_1, \dots, \mathbf{e}_p$. Then \mathbf{e}_1 points in the direction of the greatest variation in the \mathbf{u}_i , and \mathbf{e}_p in the direction of the least variation.

56.4.2 Example: Σ for the Hands Data

For example, for the data of Table 56.1,

$$\bar{\mathbf{u}} = \begin{pmatrix} 7.8975 \\ 12.7542 \\ 14.3067 \end{pmatrix}$$

$$\begin{aligned} \Sigma &= \frac{1}{12} \left\{ \left[\begin{pmatrix} 5.17 \\ 11.30 \\ 16.18 \end{pmatrix} - \bar{\mathbf{u}} \right] \left[\begin{pmatrix} 5.17 \\ 11.30 \\ 16.18 \end{pmatrix} - \bar{\mathbf{u}} \right]^T + \dots + \right. \\ &\quad \left. \left[\begin{pmatrix} 8.68 \\ 12.71 \\ 13.67 \end{pmatrix} - \bar{\mathbf{u}} \right] \left[\begin{pmatrix} 8.68 \\ 12.71 \\ 13.67 \end{pmatrix} - \bar{\mathbf{u}} \right]^T \right\} \\ &= \begin{pmatrix} 2.6249 & 1.2525 & 0.1424 \\ 1.2525 & 0.8095 & -0.5552 \\ 0.1424 & -0.5552 & 4.9306 \end{pmatrix} \end{aligned}$$

$$\lambda_1 = 5.004, \quad \lambda_2 = 3.255, \quad \lambda_3 = 0.1054,$$

$$\mathbf{e}_1 = \begin{pmatrix} -0.0115 \\ -0.1346 \\ 0.9908 \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} -0.8942 \\ -0.4420 \\ -0.0704 \end{pmatrix},$$

$$\mathbf{e}_3 = \begin{pmatrix} -0.4474 \\ 0.8869 \\ 0.1152 \end{pmatrix}$$

Examining the data of Table 56.1, one sees that $\bar{\mathbf{u}}$ is close to point G, the center of the left palm. Examining

the displacement of G to C, top of the middle finger, it is evident that left hand was close to vertically oriented. This is the direction \mathbf{e}_1 . Examining the displacement of G to E, the top of the thumb, it appears that the left thumb was pointed in roughly the direction of the x -axis. This is the direction of $-\mathbf{e}_2$. Thus, the left hand was roughly parallel to the x - z plane. The normal vector to the plane of the left hand is thus approximately parallel to the y -axis. This is the direction of \mathbf{e}_3 . Notice that λ_3 is much smaller than λ_1 or λ_2 , indicating that the thickness of the hand is much smaller than its length or breadth.

56.4.3 Statistical Assumptions for the Procrustes Model

Before giving the statistical properties of $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$, it is necessary to make explicit the statistical assumptions of the Procrustes model (56.2). These assumptions are:

- $\mathbf{u}_1, \dots, \mathbf{u}_n \in R^p$ are fixed (nonrandom) vectors.
- $\mathbf{v}_1, \dots, \mathbf{v}_n \in R^p$ are independent random vectors.
- The distribution of \mathbf{v}_i is of the form $f_0(s_i)$, where $s_i = \|\mathbf{v}_i - \gamma \mathbf{A} \mathbf{u}_i - \mathbf{b}\|$. Here $(\mathbf{A}, \gamma, \mathbf{b})$ are unknown, $\mathbf{A} \in \text{SO}(p)$ or $\text{O}(p)$, γ is a positive real constant, and $\mathbf{b} \in R^p$.

The most obvious example of a suitable distribution f_0 is

$$f_0(s) = (2\pi\sigma^2)^{-p/2} e^{-\frac{s^2}{2\sigma^2}} \quad (56.35)$$

for a fixed constant σ^2 . In what follows, we will not need to know the value of σ^2 . In fact, we will not even need to know the form of f_0 , only that the distribution of \mathbf{v}_i depends only upon its distance s_i from $\gamma \mathbf{A} \mathbf{u}_i + \mathbf{b}$.

The distribution (56.35) is a multivariate normal distribution with mean vector $\gamma \mathbf{A} \mathbf{u}_i + \mathbf{b}$ and covariance matrix $\sigma^2 \mathbf{I}_p$. Equivalently, the p components of \mathbf{v}_i are independent and each has variance σ^2 . If the components of \mathbf{v}_i were to have different variances, then the distribution of \mathbf{v}_i would not satisfy the Procrustes model assumptions.

In essence we assume that \mathbf{v}_i is isotropically (i.e., that all directions are the same) distributed around its mean vector.

56.4.4 Theorem (Distribution of $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$ for the Procrustes Model)

Suppose $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$ minimize an objective function of the form (56.13). Let $\boldsymbol{\beta} = \gamma \mathbf{A} \bar{\mathbf{u}} + \mathbf{b}$ and $\hat{\boldsymbol{\beta}} = \hat{\gamma} \hat{\mathbf{A}} \bar{\mathbf{u}} + \hat{\mathbf{b}}$. Then

- $\hat{\mathbf{A}}, \hat{\gamma}$, and $\hat{\boldsymbol{\beta}}$ are independent.
- $\hat{\boldsymbol{\beta}}$ is distributed multivariate normal with mean $\boldsymbol{\beta}$ and covariance matrix $\frac{k}{n} \mathbf{I}_p$.
- If $p = 2$, write $\hat{\mathbf{A}} = \mathbf{A} \Phi_2(\hat{\mathbf{h}})$, for $\hat{\mathbf{h}} \in R^1$. Then $\hat{\mathbf{h}}$ is normally distributed with mean 0 and variance $\frac{k}{n \text{Tr}(\boldsymbol{\Sigma})}$.
- If $p = 3$, write $\hat{\mathbf{A}} = \mathbf{A} \Phi_3(\hat{\mathbf{h}})$, for $\hat{\mathbf{h}} \in R^3$. Let $\boldsymbol{\Sigma} = \lambda_1 \mathbf{e}_1 \mathbf{e}_1^T + \lambda_2 \mathbf{e}_2 \mathbf{e}_2^T + \lambda_3 \mathbf{e}_3 \mathbf{e}_3^T$ be the spectral decomposition of $\boldsymbol{\Sigma}$. Then $\hat{\mathbf{h}}$ is distributed trivariate normal with mean 0 and covariance matrix

$$\frac{k}{n} [(\lambda_2 + \lambda_3)^{-1} \mathbf{e}_1 \mathbf{e}_1^T + (\lambda_3 + \lambda_1)^{-1} \mathbf{e}_2 \mathbf{e}_2^T + (\lambda_1 + \lambda_2)^{-1} \mathbf{e}_3 \mathbf{e}_3^T].$$

- For general p , write $\hat{\mathbf{A}} = \mathbf{A} \exp(\hat{\mathbf{H}})$, where $\hat{\mathbf{H}}$ is $p \times p$ skew-symmetric. Then $\hat{\mathbf{H}}$ has a multivariate normal density proportional to $\exp\left[-\frac{n}{2k} \text{Tr}(\hat{\mathbf{H}} \boldsymbol{\Sigma} \hat{\mathbf{H}})\right]$.
- $\hat{\gamma}$ is normally distributed with mean γ and variance $\frac{k}{n \text{Tr}(\boldsymbol{\Sigma})}$.

These results are asymptotic, that is, they are large-sample approximate distributions.

The constant k is defined to be

$$k = \frac{pE[\psi(s)^2]}{E^2[\psi'(s) + (p-1)\psi(s)s^{-1}]}, \quad (56.36)$$

where $\psi(s) = \rho'_0(s)$. Thus, k can be estimated from the sample by

$$\hat{k} = \frac{np \sum_i \psi(s_i)^2}{\left\{ \sum_i [\psi'(s_i) + (p-1)\psi(s_i)s_i^{-1}] \right\}^2}, \quad (56.37)$$

where $s_i = \|\mathbf{v}_i - \hat{\gamma} \hat{\mathbf{A}} \mathbf{u}_i - \hat{\mathbf{b}}\|$.

Theorem 56.4.4 is proven in Chang and Ko [4]. (In [4], s is defined to be $s = \|\mathbf{v} - \hat{\gamma} \hat{\mathbf{A}} \mathbf{u} - \hat{\mathbf{b}}\|^2$ and this causes the formulas (56.36) and (56.37) to be written somewhat differently there.)

56.4.5 Example: A Test of $\gamma = 1$

For the hands data, the least squares estimates were given in Example 56.2.4. Table 56.2 gives the calculation of the s_i . Substituting $p = 3$, $\rho_0(s) = s^2$, $\psi(s) = 2s$ into (56.37), $\hat{k} = (3n)^{-1} \sum_i s_i^2 = 0.0860$.

To test if the two hands are the same size, we test $\gamma = 1$. Using Example 56.4.2, $\text{Tr}(\boldsymbol{\Sigma}) = 8.365$. Hence, the variance of $\hat{\gamma}$ is 0.000860 and its standard error is 0.0293. Since $\hat{\gamma} = 0.9925$, we see that $\hat{\gamma}$ is not significantly different from 1.

Table 56.2 Calculation of residual lengths for data from Table 56.1

	Predicted \hat{v}_i			Residual			s_i
	$\hat{\gamma} \hat{\mathbf{A}} \mathbf{u}_i + \hat{\mathbf{b}}$			$\mathbf{v}_i - \hat{\mathbf{v}}_i$			$\ \mathbf{v}_i - \hat{\mathbf{v}}_i\ $
A	6.148	11.687	16.868	-0.238	-0.527	-0.318	0.660
B	8.475	10.969	18.207	0.155	-0.349	0.123	0.401
C	9.620	10.962	18.654	0.470	-0.362	-0.014	0.593
D	11.080	10.457	17.769	-0.190	0.493	0.131	0.544
E	13.206	10.816	13.865	-0.236	-0.686	0.015	0.726
F	8.691	11.176	13.247	0.099	0.034	-0.077	0.129
G	10.544	10.938	13.355	0.156	0.162	0.065	0.234
H	8.501	11.594	11.046	-0.031	-0.504	0.304	0.589
I	7.449	12.225	14.178	-0.169	0.295	-0.138	0.367
J	8.062	11.908	14.649	-0.012	0.512	-0.089	0.520
K	9.198	11.904	14.730	-0.128	0.486	0.130	0.519
L	10.026	11.724	14.573	0.125	0.446	-0.133	0.481

The L_1 estimate of γ is 1.0086. To calculate the standard error of this estimate, we use $\rho_0(s) = s$ and $\psi(s) = 1$. Hence, for the L_1 estimate, (56.37) yields $\hat{k} = 0.75(n^{-1} \sum_i s_i^{-1})^{-2}$. After recomputing the s_i using L_1 estimates of $(\mathbf{A}, \gamma, \mathbf{b})$, we obtain $\hat{k} = 0.023$. Thus, the L_1 estimate of γ has a standard error of 0.0150 and this estimate is also not significantly different from 1.

Apparently, the two hands have the same size.

General statistical theory implies that if the \mathbf{v}_i were really normally distributed, the least squares estimates would be the most efficient. In other words, least squares estimates should have the smallest standard errors. Evidently this is not true for the hands data and it appears that this data is not, in fact, normally distributed.

56.4.6 Example: A Test on A

As discussed in Sect. 56.4.2, the eigenvector \mathbf{e}_3 of $\boldsymbol{\Sigma}$ is perpendicular to the plane of the left palm. It might be of interest to test if the two hands have the same orientation; that is, after reflecting the left hand in the plane perpendicular to \mathbf{e}_3 , do the fingers and thumb of the two hands point in the same directions. We formulate this hypothesis as \mathbf{H}_0 : $\mathbf{A} = \mathbf{R}_{\mathbf{e}_3}$ where $\mathbf{R}_{\mathbf{e}_3}$ is the matrix of the reflection in plane perpendicular to \mathbf{e}_3 .

$$\begin{aligned} \mathbf{R}_{\mathbf{e}_3} &= \mathbf{I}_3 - 2\mathbf{e}_3\mathbf{e}_3^T \\ &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} - 2 \begin{pmatrix} -0.4474 \\ 0.8869 \\ 0.1152 \end{pmatrix} \begin{pmatrix} -0.4474 & 0.8869 & 0.1152 \end{pmatrix}^T \\ &= \begin{pmatrix} 0.5996 & 0.7936 & 0.1031 \\ 0.7936 & -0.5731 & -0.2044 \\ 0.1031 & -0.2044 & 0.9734 \end{pmatrix}, \end{aligned}$$

$\hat{\mathbf{h}}$ is defined by

$$\begin{aligned} \Phi_3(\hat{\mathbf{h}}) &= \mathbf{R}_{\mathbf{e}_3}^T \hat{\mathbf{A}} \\ &= \begin{pmatrix} 0.7843 & -0.6127 & -0.0976 \\ 0.5999 & 0.7890 & -0.1327 \\ 0.1583 & 0.0455 & 0.9863 \end{pmatrix}, \end{aligned} \quad (56.38)$$

where $\hat{\mathbf{A}}$ was calculated in Sect. 56.2.4.

To solve for $\hat{\mathbf{h}}$, we use the results at the end of Sect. 56.3.2. The matrix of (56.38) has an eigenvector of $\boldsymbol{\xi} = (0.1395 \ -0.2003 \ 0.9494)^T$ corresponding to the eigenvalue of 1. Its angle of rotation is given by

$$\theta = \arccos \left[0.5 \text{Tr} \left(\mathbf{R}_{\mathbf{e}_3}^T \hat{\mathbf{A}} \right) - 0.5 \right] = 0.6764.$$

Thus, $\hat{\mathbf{h}} = \theta \boldsymbol{\xi} = (0.0944 \ -0.1355 \ 0.6422)^T$.

By Theorem 56.4.4, if H_0 is true, $\hat{\mathbf{h}}$ is trivariate normally distributed with mean $\mathbf{0}$ and covariance matrix

$$\frac{k}{n} \left[(\lambda_2 + \lambda_3)^{-1} \mathbf{e}_1 \mathbf{e}_1^T + (\lambda_3 + \lambda_1)^{-1} \mathbf{e}_2 \mathbf{e}_2^T + (\lambda_1 + \lambda_2)^{-1} \mathbf{e}_3 \mathbf{e}_3^T \right].$$

The constant k was estimated in Sect. 56.4.5 and the λ_i and \mathbf{e}_i were calculated in Sect. 56.4.2. Using these calculations, the covariance matrix of $\hat{\mathbf{h}}$ is estimated to be

$$\widehat{\text{Cov}}(\hat{\mathbf{h}}) = \begin{pmatrix} 0.001296 & 0.0002134 & 0.00001923 \\ 0.0002134 & 0.0009951 & -0.0001520 \\ 0.00001923 & -0.0001520 & 0.002112 \end{pmatrix}$$

Under the null hypothesis

$$\chi^2 = \hat{\mathbf{h}}^T \widehat{\text{Cov}}(\hat{\mathbf{h}})^{-1} \hat{\mathbf{h}} = 213$$

has an approximate χ^2 distribution with three degrees of freedom.

We emphatically conclude that, after reflecting the left hand, the orientations of the two hands are not the same.

56.4.7 Asymptotic Relative Efficiency of Least Squares and L_1 Estimates

Examining Theorem 56.4.4, we see that the covariance of the M -estimate $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$ is determined, up to a constant k , by the geometry of the \mathbf{u}_i , as summarized by the matrix Σ . Only the constant k , see (56.36), depends upon the probability distribution of the \mathbf{v}_i and the objective function (56.13) that $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$ minimize. Furthermore, a sample estimate of k , see (56.37) is available which does not require knowledge of the distribution of the \mathbf{v}_i .

Let $k(f_0, L_2)$ denote the constant k as defined in (56.36) when the underlying density is of the form f_0 and least squares (L_2) estimation is used, and $k(f_0, L_1)$ the corresponding value when L_1 estimation is used. The ratio $\text{ARE}(L_1, L_2; f_0) = k(f_0, L_2)/k(f_0, L_1)$ is called the *asymptotic relative efficiency* of the L_1 to the least squares estimators at the density f_0 .

We see that

$$\text{ARE}(L_1, L_2; f_0) = \frac{\text{Variance of least squares estimator}}{\text{Variance of } L_1 \text{ estimator}}, \quad (56.39)$$

where we recognize that both variances are matrices, but the two variance matrices are multiples of each other.

If f_0 is a p -dimensional normal density (56.35), it can be shown from (56.36) that

$$\text{ARE}(L_1, L_2; N_p) = \frac{2\Gamma^2[(p+1)/2]}{p\Gamma^2(p/2)}. \quad (56.40)$$

We have used N_p in (56.40) to denote the p -dimensional normal density function.

The Γ function in (56.40) has the properties

$$\begin{aligned} \Gamma(1) &= 1 & \Gamma(0.5) &= \sqrt{\pi} \\ \Gamma(q+1) &= q\Gamma(q). \end{aligned}$$

Thus, when $p = 2, 3$

$$\begin{aligned} \text{ARE}(L_1, L_2; N_2) &= \frac{\pi}{4} = 0.785, \\ \text{ARE}(L_1, L_2; N_3) &= \frac{8}{3\pi} = 0.849. \end{aligned} \quad (56.41)$$

$\text{ARE}(L_1, L_2; N_p)$ increases to 1 as $p \rightarrow \infty$.

When the underlying distribution is normal, statistical theory indicates that least squares procedures are optimal, that

is, they have the smallest variance. Using (56.39) and (56.41), we see that for $p = 3$, even when the data is normal, the use of L_1 methods results in only an 8% penalty in standard error. And L_1 methods offer superior resistance to outliers.

Indeed, as we saw in Example 56.4.5, the standard error of the L_1 estimator was *smaller* than the standard error of the least squares estimator. Evidently the hands dataset is long-tailed, that is, it has more outliers than would be expected with normal data.

56.4.8 The Geometry of the Landmarks and the Errors in $\hat{\mathbf{A}}$

In this section, we will constrain our discussion to the case $p = 3$.

Suppose we write the estimate $\hat{\mathbf{A}}$ in the form

$$\hat{\mathbf{A}} = \mathbf{A}\Phi_3(\hat{\mathbf{h}}). \quad (56.42)$$

$\Phi_3(\hat{\mathbf{h}})$ is a (hopefully) small rotation which expresses the deviation of the estimate $\hat{\mathbf{A}}$ from the true value \mathbf{A} .

Recall that $\Phi_3(\hat{\mathbf{h}})$ is a rotation of $\|\hat{\mathbf{h}}\|$ radians around the axis $\|\hat{\mathbf{h}}\|^{-1}\hat{\mathbf{h}}$.

In particular $\Phi_3(\hat{\mathbf{h}})^{-1} = \Phi_3(-\hat{\mathbf{h}})$ and

$$\mathbf{A} = \hat{\mathbf{A}}\Phi_3(-\hat{\mathbf{h}}).$$

According to Theorem 56.4.4, the covariance matrix of $\hat{\mathbf{h}}$ has the form

$$\begin{aligned} \text{Cov}(\hat{\mathbf{h}}) &= \frac{k}{n} [(\lambda_2 + \lambda_3)^{-1} \mathbf{e}_1 \mathbf{e}_1^T + (\lambda_3 + \lambda_1)^{-1} \mathbf{e}_2 \mathbf{e}_2^T \\ &\quad + (\lambda_1 + \lambda_2)^{-1} \mathbf{e}_3 \mathbf{e}_3^T], \end{aligned} \quad (56.43)$$

where $\lambda_1 \geq \lambda_2 \geq \lambda_3$ are the eigenvalues of Σ with corresponding eigenvectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$. Since $\hat{\mathbf{h}}$ is normally distributed

$$\chi^2 = \hat{\mathbf{h}}^T \left[\text{Cov}(\hat{\mathbf{h}})^{-1} \right] \hat{\mathbf{h}}$$

is distributed χ^2 with three degrees of freedom.

Thus, a confidence region for \mathbf{A} is of the form

$$\left\{ \hat{\mathbf{A}}\Phi_3(-\hat{\mathbf{h}}) \mid \hat{\mathbf{h}}^T \left[\text{Cov}(\hat{\mathbf{h}})^{-1} \right] \hat{\mathbf{h}} < \chi_{3,\alpha}^2 \right\}, \quad (56.44)$$

where $\chi_{3,\alpha}^2$ is the appropriate critical point of a χ_3^2 distribution.

Let $\theta = \|\hat{\mathbf{h}}\|$ and $\boldsymbol{\xi} = -\|\hat{\mathbf{h}}\|^{-1}\hat{\mathbf{h}}$ so that $\hat{\mathbf{h}} = -\theta\boldsymbol{\xi}$.

Thus, $\Phi_3(\hat{\mathbf{h}})$ is a rotation of θ radians around the axis $\boldsymbol{\xi}$.

Substituting (56.43) into the confidence region (56.44), we can re-express this confidence region as

$$\left\{ \hat{\mathbf{A}} \Phi_3(\theta\boldsymbol{\xi}) \mid \theta^2 \frac{n}{k} \left[(\lambda_2 + \lambda_3) (\boldsymbol{\xi}^T \mathbf{e}_1)^2 + (\lambda_3 + \lambda_1) (\boldsymbol{\xi}^T \mathbf{e}_2)^2 + (\lambda_1 + \lambda_2) (\boldsymbol{\xi}^T \mathbf{e}_3)^2 \right] < \chi_{3,\alpha}^2 \right\}. \quad (56.45)$$

Now

$$\lambda_2 + \lambda_3 \leq \lambda_3 + \lambda_1 \leq \lambda_1 + \lambda_2.$$

Thus, the confidence region (56.45) constrains θ the most (i.e., the limits on θ are the smallest) when $\boldsymbol{\xi}$ points in the direction \mathbf{e}_3 . It bounds θ the least when $\boldsymbol{\xi}$ points in the direction \mathbf{e}_1 .

Recall also that \mathbf{e}_1 is the direction of the greatest variation in the \mathbf{u}_i and \mathbf{e}_3 the direction of the least variation.

For the hands data of Table 56.1, \mathbf{e}_1 points in the direction of the length of the left hand and \mathbf{e}_3 in the normal direction to the palm.

Thus, the angle θ of the small rotation $\Phi_3(\theta\boldsymbol{\xi})$ is the most constrained when its axis $\boldsymbol{\xi}$ points in the direction of the least variation in the \mathbf{u}_i . θ is least constrained when $\boldsymbol{\xi}$ points in the direction of the greatest variation of the \mathbf{u}_i .

For the hands data, if $\hat{\mathbf{h}}$ is in the direction of \mathbf{e}_1 , the length of the hand, it represents a small rotation at the elbow with the wrist held rigid. The variance of the deviation rotation $\hat{\mathbf{h}}$ in the direction \mathbf{e}_1 is $(\lambda_2 + \lambda_3)^{-1} = 0.298$. If $\hat{\mathbf{h}}$ points in the direction of \mathbf{e}_2 , the width of the hand, it represents a forwards and backwards rotation at the wrist; the variance of $\hat{\mathbf{h}}$ in this direction is $(\lambda_3 + \lambda_1)^{-1} = 0.196$. Finally if $\hat{\mathbf{h}}$ points in the direction of \mathbf{e}_3 , the normal vector to the hand, it represents a somewhat awkward sideways rotation at the wrist (this rotation is represented in Fig. 56.1b); the variance of $\hat{\mathbf{h}}$ in this direction is $(\lambda_1 + \lambda_2)^{-1} = 0.121$. If the variability of the component of $\hat{\mathbf{h}}$ in the direction of a rotation at the elbow is unacceptably large, we could increase λ_3 , in effect to create, if possible, landmarks which effectively thicken the palm.

A heuristic derivation of this result is due to *Stock and Molnar* [9, 10]. It appeared in the geophysical literature and is considered a major development in our understanding of the uncertainties in tectonic plate reconstructions. We will present their argument below, suitably modified for the image registration context.

It is convenient to rewrite the model, as in Theorem 56.4.4, in the form (56.29). If we substitute $\mathbf{A} = \hat{\mathbf{A}}\Phi_3(\theta\boldsymbol{\xi})$, we see that \mathbf{A} first perturbs the $\mathbf{u}_i - \bar{\mathbf{u}}$ by the small rotation $\Phi_3(\theta\boldsymbol{\xi})$ and then applies the best fitting orthogonal matrix $\hat{\mathbf{A}}$.

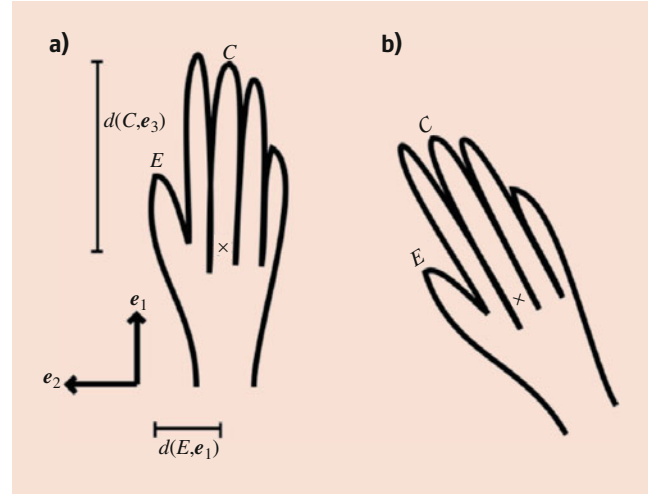


Fig. 56.1 (a) A hand with axes \mathbf{e}_1 , \mathbf{e}_2 ; axis \mathbf{e}_3 points out of paper. X marks the center point $\bar{\mathbf{u}}$. The distances $d(C, \mathbf{e}_3)$ and $d(E, \mathbf{e}_1)$ are the lengths of the indicated line segments. (b) The effect of a rotation of angle θ around the axis \mathbf{e}_3 . The point C moves a distance of approximately $d(C, \mathbf{e}_3)\theta$. Under a rotation of θ around \mathbf{e}_1 (not shown), the point E moves a distance of approximately $d(E, \mathbf{e}_1)\theta$. Notice that $d(E, \mathbf{e}_1) < d(C, \mathbf{e}_3)$, and, indeed, the landmarks \mathbf{u}_i tend to be closer to \mathbf{e}_1 than to \mathbf{e}_3 . It follows that a rotation of θ around \mathbf{e}_3 will move the figure more than a rotation of θ around \mathbf{e}_1 .

Let $d(\mathbf{u}_i, \boldsymbol{\xi})$ be the distance of the landmark \mathbf{u}_i to the line through the center point $\bar{\mathbf{u}}$ and in the direction of the axis $\boldsymbol{\xi}$. Refer to Fig. 56.1. Since the landmarks vary most in the direction \mathbf{e}_1 and least in the direction \mathbf{e}_3 , the distances $d(\mathbf{u}_i, \mathbf{e}_3)$ will tend to be biggest and the distances $d(\mathbf{u}_i, \mathbf{e}_1)$ smallest.

A point \mathbf{x} will move a distance of approximately $d(\mathbf{x}, \boldsymbol{\xi})\theta$ under a rotation of angle θ around the axis $\boldsymbol{\xi}$. It follows that a rotation of angle θ will most move the landmarks \mathbf{u}_i if the axis is \mathbf{e}_3 . It will move the landmarks \mathbf{u}_i least if the axis is \mathbf{e}_1 . In other words, for a fixed θ , the small rotation $\Phi_3(\theta\boldsymbol{\xi})$ will most degrade the best fit, provided by $\hat{\mathbf{A}}$, if $\boldsymbol{\xi} = \mathbf{e}_3$; it will least degrade the best fit if $\boldsymbol{\xi} = \mathbf{e}_1$.

An orthogonal transformation $\mathbf{A} = \hat{\mathbf{A}}\Phi_3(\theta\boldsymbol{\xi})$ is considered a possible transformation if it does not degrade the best fit by too much. It follows that θ is most constrained if $\boldsymbol{\xi} = \mathbf{e}_3$, the direction of the least variation in the landmarks \mathbf{u}_i , and is least constrained if $\boldsymbol{\xi} = \mathbf{e}_1$, the direction of greatest variation in the landmarks \mathbf{u}_i .

Suppose instead we were to write the estimate $\hat{\mathbf{A}}$ in the form

$$\begin{aligned} \hat{\mathbf{A}} &= \Phi_3(\hat{\mathbf{h}}_v) \mathbf{A}, \\ \mathbf{A} &= \Phi_3(-\hat{\mathbf{h}}_v) \hat{\mathbf{A}}. \end{aligned} \quad (56.46)$$

Then (56.43) is replaced by

$$\begin{aligned} \text{Cov}(\hat{\mathbf{h}}_v) &= \frac{k}{n} [(\lambda_2 + \lambda_3)^{-1} (\mathbf{A}\mathbf{e}_1) (\mathbf{A}\mathbf{e}_1)^T \\ &\quad + (\lambda_3 + \lambda_1)^{-1} (\mathbf{A}\mathbf{e}_2) (\mathbf{A}\mathbf{e}_2)^T \\ &\quad + (\lambda_1 + \lambda_2)^{-1} (\mathbf{A}\mathbf{e}_3) (\mathbf{A}\mathbf{e}_3)^T]. \end{aligned}$$

The same reasoning then expresses the errors of $\hat{\mathbf{h}}_v$, and hence of $\hat{\mathbf{A}}$, in terms of the geometry of the landmarks \mathbf{v}_i . In other words, for the hands data, using the definition (56.46) expresses the errors of $\hat{\mathbf{A}}$ in terms of the orientation of the right hand.

56.4.9 Statistical Properties of M -Estimates for Spherical Regressions

The statistical assumptions of the spherical regression model (56.1) are:

- $\mathbf{u}_1, \dots, \mathbf{u}_n \in \Omega_p$ are fixed (nonrandom) vectors.
- $\mathbf{v}_1, \dots, \mathbf{v}_n \in \Omega_p$ are independent random vectors.
- The distribution of \mathbf{v}_i is of the form $f_0(t_i)$ where $t_i = \mathbf{v}_i^T \mathbf{A} \mathbf{u}_i$. Here $\mathbf{A} \in \text{SO}(p)$ or $O(p)$ is unknown.

A commonly used distribution for spherical data $\mathbf{x} \in \Omega_p$ is the distribution whose density (with respect to surface measure, or uniform measure, on Ω_p) is

$$f(\mathbf{x}; \boldsymbol{\theta}) = c(\kappa) \exp(\kappa \mathbf{x}^T \boldsymbol{\theta}). \quad (56.47)$$

This distribution has two parameters: a positive real constant κ which is commonly called the *concentration parameter* and $\boldsymbol{\theta} \in \Omega_p$. It is easily seen that $f(\mathbf{x})$ is maximized over $\mathbf{x} \in \Omega_p$ at $\boldsymbol{\theta}$ and hence $\boldsymbol{\theta}$ is usually referred to as the *modal vector*; $c(\kappa)$ is a normalizing constant.

If $\kappa = 0$, (56.47) is a uniform density on Ω_p . On the other hand as $\kappa \rightarrow \infty$, the density (56.47) approaches that of a multivariate normal distribution in $p - 1$ dimensions with a covariance matrix of $\kappa^{-1} \mathbf{I}_{p-1}$. Thus, intuitively we can think of κ as σ^{-2} , that is, think of κ as the inverse variance. As $\kappa \rightarrow \infty$, (56.47) approaches a singular multivariate normal distribution supported on the $(p - 1)$ -dimensional subspace $\boldsymbol{\theta}^\perp \subset R^p$. As a singular multivariate, normal distribution in R^p its covariance matrix is $\kappa^{-1} (\mathbf{I}_p - \boldsymbol{\theta} \boldsymbol{\theta}^T)$.

For the circle Ω_1 , (56.47) is due to von Mises. For general Ω_p , it is due (independently) to Fisher and to Langevin. More properties of the Fisher–von Mises–Langevin distribution can be found in Watson [11] or in Fisher et al. [12].

The distribution of an M -estimator $\hat{\mathbf{A}}$ which minimizes an objective function of the form (56.14) is similar to the distribution given in Theorem 56.4.4:

- If $p = 2$, write $\hat{\mathbf{A}} = \mathbf{A} \Phi_2(\hat{\mathbf{h}})$, for $\hat{\mathbf{h}} \in R^1$. Then $\hat{\mathbf{h}}$ is normally distributed with mean 0 and variance $\frac{k}{n}$.

- If $p = 3$, write $\hat{\mathbf{A}} = \mathbf{A} \Phi_3(\hat{\mathbf{h}})$, for $\hat{\mathbf{h}} \in R^3$. Let $\boldsymbol{\Sigma} = \lambda_1 \mathbf{e}_1 \mathbf{e}_1^T + \lambda_2 \mathbf{e}_2 \mathbf{e}_2^T + \lambda_3 \mathbf{e}_3 \mathbf{e}_3^T$ be the spectral decomposition of $\boldsymbol{\Sigma}$. Then $\hat{\mathbf{h}}$ is distributed trivariate normal with mean $\mathbf{0}$ and covariance matrix

$$\begin{aligned} \frac{k}{n} [(\lambda_2 + \lambda_3)^{-1} \mathbf{e}_1 \mathbf{e}_1^T + (\lambda_3 + \lambda_1)^{-1} \mathbf{e}_2 \mathbf{e}_2^T \\ + (\lambda_1 + \lambda_2)^{-1} \mathbf{e}_3 \mathbf{e}_3^T]. \end{aligned}$$

- For general p , write $\hat{\mathbf{A}} = \mathbf{A} \exp(\hat{\mathbf{H}})$, where $\hat{\mathbf{H}}$ is $p \times p$ skew-symmetric. Then $\hat{\mathbf{H}}$ has a multivariate normal density proportional to $\exp\left[-\frac{n}{2k} \text{Tr}(\hat{\mathbf{H}}^T \boldsymbol{\Sigma} \hat{\mathbf{H}})\right]$.

Let $\psi(t) = -\rho'_0(t)$. (The sign of ψ has been chosen to make $\psi(t)$ nonnegative, since ρ_0 is a decreasing function of t .) The constant k and its sample estimate \hat{k} are given by

$$\begin{aligned} k &= \frac{(p-1)E[\psi(t)^2(1-t^2)]}{E^2[(p-1)\psi(t)t - \psi'(t)(1-t^2)]}, \\ \hat{k} &= \frac{n(p-1)\sum_i \psi(t_i)^2(1-t_i^2)}{\left\{\sum_i [(p-1)\psi(t_i)t_i - \psi'(t_i)(1-t_i^2)]\right\}^2}. \end{aligned} \quad (56.48)$$

For the spherical case, the matrix $\boldsymbol{\Sigma} = \sum_i \mathbf{u}_i \mathbf{u}_i^T$. Its dominant eigenvector \mathbf{e}_1 points in the direction of the center of the \mathbf{u}_i . The \mathbf{e}_2 is the vector perpendicular to \mathbf{e}_1 so that the two-dimensional plane spanned by \mathbf{e}_1 and \mathbf{e}_2 (and the origin) best fits the \mathbf{u}_i . This continues until $\mathbf{e}_1, \dots, \mathbf{e}_{p-1}$ is the $(p - 1)$ -dimensional hyperplane, among the collection of all $(p - 1)$ -dimensional hyperplanes that best fits the data. This latter hyperplane is, of course, the hyperplane perpendicular to \mathbf{e}_p . Except for this slight reinterpretation of the geometric meaning of the \mathbf{e}_i , our previous comments about the relationship of the uncertainties in $\hat{\mathbf{h}}$ to the geometry of the \mathbf{u} -points, as summarized by the eigen-decomposition of $\boldsymbol{\Sigma}$, remain valid. Indeed the original Stock and Molnar insights about the uncertainties of tectonic plate reconstructions were actually in the spherical data context.

Thus, as before, the uncertainties in $\hat{\mathbf{A}}$ are determined up to the constant k by the geometry of the \mathbf{u} -points. Only the constant k depends upon the underlying data distribution f_0 or upon the objective function ρ . We can define the asymptotic relative efficiency as in Sect. 56.4.7 without change. Its interpretation (56.39) also remains valid.

Equation (56.48) implies that we can, as before, define the asymptotic efficiency of the L_1 estimator relative to the least squares estimator, at the density f_0 , as $\text{ARE}(L_1, L_2; f_0) = k(f_0, L_2)/k(f_0, L_1)$. The interpretation (56.39) remains valid. The constants $k(f_0, L_2)$ and $k(f_0, L_1)$ come from (56.48) using the underlying density f_0 under consideration and $\rho_0(t) = 2 - 2t$ [refer to (56.9)], $\psi(t) = 2$, for the least squares case, or $\rho_0(t) = \arccos(t)$, $\psi(t) = (1 - t^2)^{\frac{1}{2}}$, for the L_1 case. If f_0 is the Fisher–von Mises–Langevin density (56.47) on Ω_p (which we will denote by $F_{\kappa,p}$ in the following)

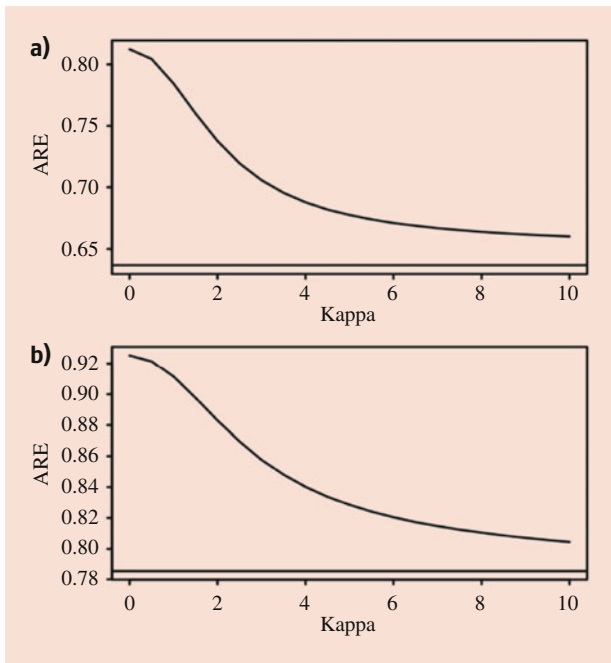


Fig. 56.2 Asymptotic efficiency of L_1 estimators relative to least squares estimators for Fisher–von Mises–Langevin distributions on Ω_p as a function of κ for (a) $p = 2$ and (b) $p = 3$. Horizontal lines are asymptotic limits as $\kappa \rightarrow \infty$.

$$\begin{aligned} \text{ARE}(L_1, L_2; F_{\kappa,p}) &= \frac{\left[\int_{-1}^1 e^{\kappa t} (1-t^2)^{(p-2)/2} dt \right]^2}{\left[\int_{-1}^1 e^{\kappa t} (1-t^2)^{(p-1)/2} dt \right]^2} \frac{1}{\left[\int_{-1}^1 e^{\kappa t} (1-t^2)^{(p-3)/2} dt \right]} \end{aligned} \tag{56.49}$$

As $\kappa \rightarrow \infty$, the limit of (56.49) is

$$\begin{aligned} \lim_{\kappa \rightarrow \infty} \text{ARE}(L_1, L_2; F_{\kappa,p}) &= \frac{2\Gamma^2(p/2)}{(p-1)\Gamma^2[(p-1)/2]} \end{aligned} \tag{56.50}$$

Comparing (56.40) with (56.50), we see that (56.50) is the same as (56.40) with p replaced by $p - 1$. This is as expected because, as noted above, for large κ the Fisher–von Mises–Langevin distribution approaches a $(p - 1)$ -dimensional multivariate normal distribution. Figure 56.2 gives a graph of $\text{ARE}(L_1, L_2; F_{\kappa,p})$ for $p = 2, 3$.

In particular for $p = 3$, $\text{ARE}(L_1, L_2; F_{\kappa,3}) \rightarrow \pi/4$. For the Fisher–von–Mises–Langevin distribution, least squares methods are optimal. Nevertheless, in standard error terms, the penalty for using L_1 methods is at most 13%.

56.5 Diagnostics

We discuss in this section influence function diagnostics for the Procrustes model. Suppose the registration provided by the estimates $(\hat{\mathbf{A}}, \hat{\gamma}, \hat{\mathbf{b}})$ is unsatisfactory. These diagnostics

will determine which points are influential for the estimated orthogonal matrix $\hat{\mathbf{A}}$, which points are influential for the estimated scale change $\hat{\gamma}$, and which are influential for the estimated translation $\hat{\mathbf{b}}$.

56.5.1 Influence Diagnostics in Simple Linear Regression

As background discussion, we consider first the simple linear regression model

$$y_i = \alpha + \beta x_i + \text{error}, \tag{56.51}$$

where $x_i, y_i \in R^1$. For simplicity, we will assume $\sum_i x_i = 0$. This can be accomplished by a centering transformation similar to that used in (56.29).

For the model (56.51), the least squares estimates are

$$\begin{aligned} \hat{\alpha} &= \bar{y}, \\ \hat{\beta} &= \left(\sum_i x_i^2 \right)^{-1} \left(\sum_i x_i y_i \right). \end{aligned} \tag{56.52}$$

Suppose we delete the i -th observation (x_i, y_i) and recompute the estimates (56.52). The resulting estimates would be [see Cook and Weisberg [13], (Sect. 3.4.6)]

$$\begin{aligned} \hat{\alpha}_{(i)} &= \hat{\alpha} - (1 - v_{ii})^{-1} \frac{e_i}{n}, \\ \hat{\beta}_{(i)} &= \hat{\beta} - (1 - v_{ii})^{-1} \frac{x_i e_i}{\sum_k x_k^2}, \end{aligned} \tag{56.53}$$

where

$$e_i = y_i - \hat{\alpha} - \hat{\beta} x_i$$

is the residual, and

$$v_{ii} = \frac{1}{n} + \frac{x_i^2}{\sum_k x_k^2}$$

is the i -th diagonal entry of the so-called *hat* matrix. It can be shown that

$$\begin{aligned} 0 \leq v_{ii} &\leq 1, \\ \sum_i v_{ii} &= 2. \end{aligned} \tag{56.54}$$

If $|x_i|$ is big, $1 - v_{ii}$ can be close to zero, although because of (56.54), if n is large, this will usually not be the case. Ignoring the factor of $(1 - v_{ii})^{-1}$, it follows from (56.53) that deletion of (x_i, y_i) will be influential for $\hat{\alpha}$ when the magnitude of the residual $|e_i|$ is big. Deletion of (x_i, y_i) will be influential for $\hat{\beta}$ when both $|x_i|$ and $|e_i|$ are big. Points with large values of $|x_i|$ [typically, due to (56.54), $|x_i| > \frac{4}{n}$] are

called *high-leverage* points, whereas points with large values of $|e_i|$ are called *outliers*. (Recall we have centered the data so that $\bar{x} = 0$.)

Thus, influence on $\hat{\alpha}$ and on $\hat{\beta}$ is different. Outliers are influential for $\hat{\alpha}$, whereas influence for $\hat{\beta}$ is a combination of being an outlier and having high leverage. For the model (56.51) with the least squares estimators, the *influence function* works out to be

$$\text{IF}[\hat{\alpha}; (x_i, y_i)] = \frac{y_i - \alpha - \beta x_i}{n}, \quad (56.55)$$

$$\text{IF}[\hat{\beta}; (x_i, y_i)] = \frac{x_i (y_i - \alpha - \beta x_i)}{\sum_k x_k^2}, \quad (56.56)$$

where α and β are the “true” population values in the model (56.51). We will not give a formal definition of the influence function here, but refer the reader to *Cook and Weisberg [13]* for a more comprehensive discussion of the influence function in the regression model.

It should be noted that to actually calculate (56.55) and (56.56) from a sample, it is necessary to estimate α and β . Thus, even though in the left-hand sides of (56.55) and (56.56), $\hat{\alpha}$ and $\hat{\beta}$ are least squares estimates, we should substitute in the right-hand sides of (56.55) and (56.56) better estimates, if available, of α and β . There is no contradiction in using L_1 estimates to estimate the influence function of the least squares estimators.

56.5.2 Influence Diagnostics for the Procrustes Model

Chang and Ko [4] calculated the *standardized influence functions* (SIF) for M -estimates (56.13) in the Procrustes model (56.29). (The influence functions of the estimates $\hat{\mathbf{A}}$ and $\hat{\beta}$ are vectors; the standardization calculates their square length in some metric.) Using their notation

$$\left\| \text{SIF}[\hat{\beta}; (\mathbf{u}_i, \mathbf{v}_i)] \right\|^2 = k_I \psi(s_i)^2, \quad (56.57)$$

where $s_i = \|\mathbf{v}_i - \gamma \mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) - \beta\|$ and $\psi(s) = \rho'_0(s)$. Therefore

- The influence of $(\mathbf{u}_i, \mathbf{v}_i)$ on the estimate $\hat{\beta}$ of the translation parameter depends only upon the length s_i of the residual.

This behavior is similar to that of simple linear regression (56.55). The constant k_I is given by

$$k_I = \frac{pE[g'(s)^2]}{E^2[\psi'(s) + (p-1)\psi(s)s^{-1}]},$$

where $g(s) = \log f_0(s)$ and $f_0(s)$ is defined in Sect. 56.4.3.

For the scale parameter γ , let $\Sigma = n^{-1} \sum_i (\mathbf{u}_i - \bar{\mathbf{u}})(\mathbf{u}_i - \bar{\mathbf{u}})^T$. Then

$$\left\| \text{SIF}[\hat{\gamma}; (\mathbf{u}_i, \mathbf{v}_i)] \right\|^2 = \frac{k_I \psi(s_i)^2}{\text{Tr}(\Sigma)} [\mathbf{w}_i^T \mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}})]^2. \quad (56.58)$$

Here

$$\mathbf{w}_i = [\mathbf{v}_i - \gamma \mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) - \beta] / s_i.$$

Notice that $\mathbf{v}_i - \gamma \mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}}) - \beta$ is the residual of the i -th data point and s_i is its length. Thus, \mathbf{w}_i is a unit-length vector in the direction of the i -th data point residual. We conclude

- For a given length s_i of residual, a point $(\mathbf{u}_i, \mathbf{v}_i)$ will be influential for the estimate $\hat{\gamma}$ of the scale parameter if \mathbf{u}_i is far from the center $\bar{\mathbf{u}}$ of the data and if its residual is parallel to $\mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}})$.

For simplicity, we restrict the formulas of influence on the estimate of the orthogonal matrix \mathbf{A} to the cases $p = 2, 3$. For $p = 2$,

$$\left\| \text{SIF}[\hat{\mathbf{A}}; (\mathbf{u}_i, \mathbf{v}_i)] \right\|^2 = \frac{k_I \psi(s_i)^2}{\text{Tr}(\Sigma)} \|\mathbf{w}_i \times [\mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}})]\|^2 \quad (56.59)$$

The product on the right-hand side of (56.59) is the vector “cross” product. Therefore

- For $p = 2$, for a given length s_i of residual, a point $(\mathbf{u}_i, \mathbf{v}_i)$ will be influential for the estimate $\hat{\mathbf{A}}$ of the orthogonal matrix if \mathbf{u}_i is far from the center $\bar{\mathbf{u}}$ of the data and if its residual is perpendicular to $\mathbf{A}(\mathbf{u}_i - \bar{\mathbf{u}})$. Thus, points that are influential for $\hat{\mathbf{A}}$ will not be influential for $\hat{\gamma}$, and vice versa. Indeed

$$\begin{aligned} \left\| \text{SIF}[\hat{\gamma}; (\mathbf{u}_i, \mathbf{v}_i)] \right\|^2 + \left\| \text{SIF}[\hat{\mathbf{A}}; (\mathbf{u}_i, \mathbf{v}_i)] \right\|^2 \\ = \frac{k_I \psi(s_i)^2}{\text{Tr}(\Sigma)} \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2. \end{aligned}$$

For $p = 3$, let $\lambda_1 \geq \lambda_2 \geq \lambda_3$ be the eigenvalues of Σ and let $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ be the corresponding eigenvectors. Write

$$\mathbf{w}_i \times \left(\mathbf{A} \frac{\mathbf{u}_i - \bar{\mathbf{u}}}{\|\mathbf{u}_i - \bar{\mathbf{u}}\|} \right) = x_1 \mathbf{A} \mathbf{e}_1 + x_2 \mathbf{A} \mathbf{e}_2 + x_3 \mathbf{A} \mathbf{e}_3.$$

Then

$$\begin{aligned} \left\| \text{SIF}[\hat{\mathbf{A}}; (\mathbf{u}_i, \mathbf{v}_i)] \right\|^2 &= k_I \psi(s_i)^2 \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2 \\ &\left(\frac{x_1^2}{\lambda_2 + \lambda_3} + \frac{x_2^2}{\lambda_3 + \lambda_1} + \frac{x_3^2}{\lambda_1 + \lambda_2} \right) \end{aligned} \quad (56.60)$$

It follows

- For $p = 3$, for a given length s_i of residual and distance $\|\mathbf{u}_i - \bar{\mathbf{u}}\|$ of \mathbf{u}_i from the center of the data, a point $(\mathbf{u}_i, \mathbf{v}_i)$ will be maximally influential for the estimate $\hat{\mathbf{A}}$ of the orthogonal matrix if both $\mathbf{u}_i - \bar{\mathbf{u}}$ is perpendicular to the dominant eigenvector \mathbf{e}_1 of Σ and the residual

$$\mathbf{w}_i = \pm \mathbf{A} \left(\frac{\mathbf{u}_i - \bar{\mathbf{u}}}{\|\mathbf{u}_i - \bar{\mathbf{u}}\|} \times \mathbf{e}_1 \right).$$

- The influence of $(\mathbf{u}_i, \mathbf{v}_i)$ on $\hat{\mathbf{A}}$ will be zero if

$$\mathbf{w}_i = \pm \hat{\mathbf{A}} \left(\frac{\mathbf{u}_i - \bar{\mathbf{u}}}{\|\mathbf{u}_i - \bar{\mathbf{u}}\|} \right).$$

- The maximum influence of the data on the estimate $\hat{\mathbf{A}}$ of the orthogonal matrix can be minimized for fixed $\text{Tr}(\Sigma)$ by making $\lambda_1 = \lambda_2 = \lambda_3$. Thus, the optimal choice of landmarks would make the landmarks spherically symmetric around the center point $\bar{\mathbf{u}}$.

56.5.3 Example: Influence for the Hands Data

For the Procrustes model (56.29) and the hands data, we compare here the influence statistics for the least squares estimates $(\hat{\mathbf{A}}_2, \hat{\gamma}_2, \hat{\beta}_2)$ [given in (56.20) and (56.33)] to those for the L_1 estimates $(\hat{\mathbf{A}}_1, \hat{\gamma}_1, \hat{\beta}_1)$ [in (56.34)]. These estimates correspond to $\psi_2(s) = s$ and $\psi_1(s) = 1$, respectively. In the right-hand sides of (56.57), (56.58), and (56.60), we substituted $\mathbf{v}_i - \hat{\gamma}_1 \hat{\mathbf{A}}_1 (\mathbf{u}_i - \bar{\mathbf{u}}) - \hat{\beta}_1$ to calculate the influence functions for both the L_1 and least squares estimates. Similarly the \mathbf{w}_i were calculated using the L_1 estimates. Furthermore, when (56.57), (56.58), and (56.60) were calculated for the i -th observation $(\mathbf{u}_i, \mathbf{v}_i)$, \mathbf{u}_i was not used to calculate Σ .

Using (56.57),

$$\begin{aligned} \left\| \text{SIF} \left[\hat{\beta}_2; (\mathbf{u}_i, \mathbf{v}_i) \right] \right\|^2 &\propto s_i^2, \\ \left\| \text{SIF} \left[\hat{\beta}_1; (\mathbf{u}_i, \mathbf{v}_i) \right] \right\|^2 &\propto 1, \end{aligned}$$

so that E (top of thumb), followed by H (base of palm), are the most influential for $\hat{\beta}_2$. All points are equally influential for $\hat{\beta}_1$.

In what follows we will be interested in determining which data points are most influential for which estimates. In other words, we will be interested in the relative values of $\|\text{SIF}\|^2$. Thus, for each estimator, we renormalized the values of $\|\text{SIF}\|^2$ so that their sum (over the 12 data points) equals 1. The results, together with the values of s_i , are shown in Fig. 56.3.

We have from (56.58) and (56.60)

$$\begin{aligned} \left\| \text{SIF} \left[\hat{\gamma}_2; (\mathbf{u}_i, \mathbf{v}_i) \right] \right\|^2 &\propto s_i^2 \left[\mathbf{w}_i^T \mathbf{A} (\mathbf{u}_i - \bar{\mathbf{u}}) \right]^2, \\ \left\| \text{SIF} \left[\hat{\gamma}_1; (\mathbf{u}_i, \mathbf{v}_i) \right] \right\|^2 &\propto \left[\mathbf{w}_i^T \mathbf{A} (\mathbf{u}_i - \bar{\mathbf{u}}) \right]^2, \\ \left\| \text{SIF} \left[\hat{\mathbf{A}}_2; (\mathbf{u}_i, \mathbf{v}_i) \right] \right\|^2 &\propto s_i^2 \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2 \\ &\quad \left(\frac{x_1^2}{\lambda_2 + \lambda_3} + \frac{x_2^2}{\lambda_3 + \lambda_1} + \frac{x_3^2}{\lambda_1 + \lambda_2} \right), \\ \left\| \text{SIF} \left[\hat{\mathbf{A}}_1; (\mathbf{u}_i, \mathbf{v}_i) \right] \right\|^2 &\propto \|\mathbf{u}_i - \bar{\mathbf{u}}\|^2 \\ &\quad \left(\frac{x_1^2}{\lambda_2 + \lambda_3} + \frac{x_2^2}{\lambda_3 + \lambda_1} + \frac{x_3^2}{\lambda_1 + \lambda_2} \right), \\ \mathbf{w}_i \times \left[\hat{\mathbf{A}}_1 \frac{\mathbf{u}_i - \bar{\mathbf{u}}}{\|\mathbf{u}_i - \bar{\mathbf{u}}\|} \right] &= x_1 \hat{\mathbf{A}}_1 \mathbf{e}_1 + x_2 \hat{\mathbf{A}}_1 \mathbf{e}_2 + x_3 \hat{\mathbf{A}}_1 \mathbf{e}_3. \end{aligned}$$

Examining Fig. 56.3, we see that point E is by far the most influential point for $\hat{\mathbf{A}}$. Its relative influence however can be somewhat diminished by using L_1 estimates. The value of $\|\mathbf{u}_E - \bar{\mathbf{u}}\|$ is also the largest of the $\|\mathbf{u}_i - \bar{\mathbf{u}}\|$. It turns out that $\mathbf{u}_E - \bar{\mathbf{u}}$ makes an angle of 13° with \mathbf{e}_2 and that the unit length \mathbf{w}_E makes an angle of 12° with $\hat{\mathbf{A}}_1 \mathbf{e}_3$. Thus, x_1 will be relatively big and x_2, x_3 relatively small. This accounts for the strong influence of point E on both estimates of \mathbf{A} . Notice that s_E and $\|\mathbf{u}_E - \bar{\mathbf{u}}\|$ are sufficiently big that, despite the directions of \mathbf{w}_E and $\hat{\mathbf{A}}_1 (\mathbf{u}_E - \bar{\mathbf{u}})$, E is still fairly influential for $\hat{\gamma}_2$. However, its influence on $\hat{\gamma}_1$, which does not depend upon s_E , is quite small.

The point H (base of the palm) is the most influential point for $\hat{\gamma}$. H is perhaps the least well-defined point so that it is

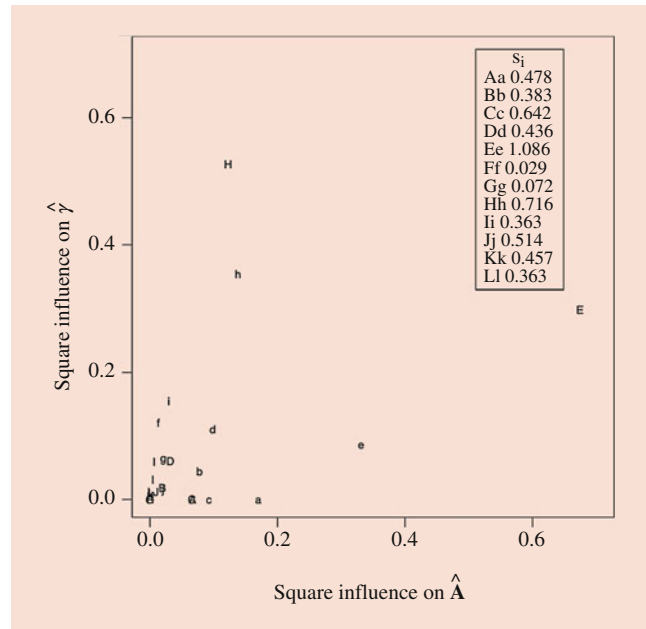


Fig. 56.3 Relative influence of hands data points on least squares (upper case) and L_1 estimates (lower case) of γ and \mathbf{A}

not surprising that its residual length s_H is relatively big. It also defines the length of the hand, so that its influence on $\hat{\gamma}$ is not surprising. Indeed if H were completely deleted, $\hat{\gamma}_2$ would change from 0.9925 to 1.0110 and $\hat{\gamma}_1$ changes from 1.0086 to 1.0262.

One might think that C (top of the middle finger) would also be influential for $\hat{\gamma}$. In a coordinate system of the eigenvectors of Σ ,

$$\mathbf{u}_H - \bar{\mathbf{u}} = [-3.98 \ 1.15 \ 0.33]^T$$

$$\mathbf{u}_C - \bar{\mathbf{u}} = [3.55 \ -0.77 \ -0.03]^T$$

so that $\mathbf{u}_H - \bar{\mathbf{u}} \approx -(\mathbf{u}_C - \bar{\mathbf{u}})$. It is useful here to remember that \mathbf{e}_1 is approximately in the direction of the length of the left hand. Furthermore, s_C and s_H are reasonably close.

However, Fig. 56.3 indicates that C has negligible influence on both estimates of γ . Indeed if C were completely deleted, $\hat{\gamma}_2$ would only change from 0.9925 to 0.9895 and $\hat{\gamma}_1$ change from 1.0086 to 1.0047. These changes are much smaller than those caused by the deletion of H.

The difference is that $\hat{\mathbf{A}}_1(\mathbf{u}_C - \bar{\mathbf{u}})$ makes an angle of 88° with \mathbf{w}_C . In other words, $\hat{\mathbf{A}}_1(\mathbf{u}_C - \bar{\mathbf{u}})$ and \mathbf{w}_C are very close to perpendicular. (Perhaps the close to perpendicularity of the residual at C to $\hat{\mathbf{A}}_1(\mathbf{u}_C - \bar{\mathbf{u}})$ is to be expected. The uncertainty in locating C is roughly tangential to the middle finger tip.) Hence, the influence of C on $\hat{\gamma}$ is negligible.

On the other hand, $\hat{\mathbf{A}}_1(\mathbf{u}_H - \bar{\mathbf{u}})$ makes an angle of 124° with \mathbf{w}_H . This accounts for the greater influence of H.

Thus, if the registration between the two hands is unsatisfactory in either the translation or rotation parameters, point E should be inspected. If it is unsatisfactory in the scale change, point H should be checked.

56.6 Location Problems and the Relationship to Regression Problems

Chang and Rivest [1] discuss the relationship between location and regression problems in a more general context. Their paper explains certain phenomenon that arises in the regression type problems explored in this chapter.

56.6.1 Location Problems Associated with Spherical and Procrustes Regressions

Ko and Chang [14] derive the large sample distribution of M -estimates for location problems on the sphere Ω_p . They assume

- $\mathbf{v}_1, \dots, \mathbf{v}_n \in \Omega_p$ are independent random vectors.
- The distribution of \mathbf{v}_i is of the form $f_0(t_i)$ where $t_i = \mathbf{v}_i^T \boldsymbol{\theta}$ for some (unknown) $\boldsymbol{\theta} \in \Omega_p$.

The relationship of these assumptions to those of the spherical regression model given in Sect. 56.4.9 is clear.

A M -estimate $\hat{\boldsymbol{\theta}}$ for $\boldsymbol{\theta}$ minimizes an objection function of the form $\rho(\boldsymbol{\theta}) = \sum_i \rho_0(t_i)$. Write $\hat{\boldsymbol{\theta}}$ in the form

$$\hat{\boldsymbol{\theta}} = \cos(\|\mathbf{h}\|) \boldsymbol{\theta} + \sin(\|\mathbf{h}\|) \|\mathbf{h}\|^{-1} \mathbf{h} \quad (56.61)$$

where $\mathbf{h} \in \boldsymbol{\theta}^\perp$, the vectors in R^p orthogonal to $\boldsymbol{\theta}$. We note that $\boldsymbol{\theta}^\perp$ consists of the vectors tangent to Ω_p at $\boldsymbol{\theta}$ and that the accompanying chapter [8] develops more completely the use of tangent space approximations in geometric statistics. It is also explained there that (56.61) is the spherical analog to the use of Φ_2, Φ_3 and the exponential map in $\text{SO}(p)$.

[14] establishes that the large sample distribution of \mathbf{h} is multivariate normal with mean $\mathbf{0}$ and covariance matrix $\frac{k}{n}(\mathbf{I}_p - \boldsymbol{\theta}\boldsymbol{\theta}^T)$ where k and its estimate \hat{k} are given in (56.48). Thus, all the previous comments about asymptotic relative efficiencies in the spherical regression model also hold for spherical location models. We note that, unlike the standard linear model, the spherical location model is not a special case of the spherical regression model.

The analogous location problem for Procrustes regression is

- $\mathbf{v}_1, \dots, \mathbf{v}_n \in R^p$ are independent random vectors.
- The distribution of \mathbf{v}_i is of the form $f_0(s_i)$, where $s_i = \|\mathbf{v}_i - \mathbf{b}\|$. Here $\mathbf{b} \in R^p$ is unknown.

The estimate $\hat{\mathbf{b}}$ minimizes the objective function $\rho(\mathbf{b}) = \sum_i \rho_0(s_i)$. Standard M -estimation theory (see for example Huber [7]) establishes that the asymptotic distribution of $\hat{\mathbf{b}} - \mathbf{b}$ is multivariate normal with mean $\mathbf{0}$ and covariance matrix $\frac{k}{n}\mathbf{I}_p$. Here k and its estimate are given by equations (56.36) and (56.37). Hence, the previous comments about relative efficiency in the Procrustes regression model also hold for these location models.

56.6.2 Location and Regression Problems in Statistical Group Models

As discussed in Sect. 56.2.1 $G = \text{SO}(p)$ is a group in the mathematical sense. (For the sake of simplicity, we will not discuss $\text{O}(p)$.) The sphere $X = \Omega_p$ is a G -space. That is, there is a map $G \times X \rightarrow X$ and one usually denotes by $g \cdot x$ the image of $(g, x) \in G \times X$ under this map. In the case of $\text{SO}(p)$ and Ω_p , this map is

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{A}\mathbf{x}. \quad (56.62)$$

Here the left hand side of (56.62) is to be interpreted, analogously to $g \cdot x$, using standard G -space notation and the right hand side of (56.62) is interpreted as matrix multiplication.

G -spaces require compatibility with the group operations

$$(g_1 * g_2) \cdot x = g_1 \cdot (g_2 \cdot x)$$

$$1 \cdot x = x.$$

Here “ $*$ ” is the group multiplication and “1” is its identity element.

Ω_p has a G -invariant inner product $\langle x, y \rangle = \langle g \cdot x, g \cdot y \rangle$. In particular

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y} = (\mathbf{A}\mathbf{x})^T (\mathbf{A}\mathbf{y}) = \langle \mathbf{A} \cdot \mathbf{x}, \mathbf{A} \cdot \mathbf{y} \rangle.$$

Notice also that under the location model in Sect. 56.6.1, the distribution of \mathbf{v}_i depends only upon $\langle \mathbf{v}_i, \theta \rangle = \mathbf{v}_i^T \theta$.

$G = \{(\mathbf{A}, \gamma, \mathbf{b})\}$ is also a group and $X = \mathbb{R}^p$ a G -space. However, Euclidean dot product is not G -invariant and, for this reason, we will use $G = \{(\mathbf{A}, \mathbf{b})\}$. This group is called the *special Euclidean* motions and denoted by $SE(p)$. Now Euclidean dot product is G -invariant; the location model in Section 56.6.1 depends only upon this inner product.

Given a G -space X and a G -invariant inner product $\langle x, y \rangle$, suppose

- $y_1, \dots, y_n \in X$ are independent random vectors.
- The distribution of \mathbf{y}_i is of the form $f_0(t_i)$, where $t_i = \langle \mathbf{y}_i, \theta \rangle$ for some unknown $\theta \in X$.

This is a location problem.

The associated regression model assumes

- $x_1, \dots, x_n \in X$ are fixed (nonrandom) vectors.
- $y_1, \dots, y_n \in X$ are independent random vectors.
- The distribution of \mathbf{y}_i is of the form $f_0(t_i)$, where $t_i = \langle \mathbf{y}_i, g \cdot x_i \rangle$ for some unknown $g \in G$.

Consider M -estimates $\hat{\theta}$ in the location problem and \hat{g} in the regression problem that minimize an objective function of the form $\sum_i \rho_0(t_i)$. *Chang and Rivest [1]* show how the large sample distribution of \hat{g} in the regression problem can be calculated from the distribution of its sibling $\hat{\theta}$ in the location problem. These results depend heavily on the tangent space approximations discussed in the accompanying chapter [8]. In particular, if a certain tangent space representation is irreducible (in the mathematical group representation sense), and appropriate Euclidean parameterizations of G and X are used

- The asymptotic covariance of \hat{g} breaks up as a product of a constant that depends only upon ρ_0 and f_0 and a matrix that depends only upon x_1, \dots, x_n .

- The asymptotic relative efficiencies between the location and regression models will be equal.

Thus, *Chang and Rivest [1]* explain some of the relationships between the Procrustes and spherical regression models and the corresponding location problems discussed in Section 56.6.1.

References

1. Chang, T., Rivest, L.-P.: M-estimation for location and regression parameters in group models: A case study using Stiefel manifolds. *Annals of Statistics*. **29**(3), 784–814 (2001)
2. Wahba, G.: Section on problems and solutions: A least squares estimate of satellite attitude. *SIAM (Society of Industrial and Applied Mathematics) Review*. **8**, 384–385 (1966)
3. Chapman, G.R., Chen, G., Kim, P.T.: Assessing geometric integrity through spherical regression techniques. *Statistica Sinica*. **5**, 173–220 (1995)
4. Chang, T., Ko, D.: M -estimates of rigid body motion on the sphere and in Euclidean space. *Annals of Statistics*. **23**, 1823–1847 (1995)
5. Colin, G.: Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society B*. **53**, 285–339 (1991)
6. Chang, T.: Spherical regression. *Annals of Statistics*. **14**, 907–924 (1986)
7. Huber, P.J.: *Robust Statistics*. Wiley, New York (1981)
8. Chang, T.: Tangent Space Approximations in Geometric Statistics: Spherical Regression and the Statistics of Tectonic Plate Reconstructions. In: Pham, H. (ed.) *Springer Handbook of Engineering Statistics*, 2nd edn. Springer-Verlag, London (2023)
9. Stock, J., Molnar, P.: Some geometrical aspects of uncertainties in combined plate reconstructions. *Geology*. **11**, 697–701 (1983)
10. Molnar, P., Stock, J.: A method for bounding uncertainties in combined plate reconstructions. *Journal of Geophysical Research*. **90**, 12537–12544 (1985)
11. Watson, G.S.: *Statistics on Spheres*. Wiley Interscience, New York (1983)
12. Fisher, N.I., Lewis, T., Embleton, B.J.J.: *Statistical Analysis of Spherical Data*. Cambridge University Press, Cambridge (1987)
13. Cook, R.D., Weisberg, S.: *Residuals and Influence in Regression*. Chapman Hall, New York (1982)
14. Ko, D., Chang, T.: Robust M -estimators on spheres. *Journal of Multivariate Analysis*. **45**, 104–136 (1993)



Ted Chang received his Ph.D. in mathematics (algebraic topology) from the University of California, Berkeley in 1972. He is currently an Emeritus Professor of Statistics at the University of Virginia. His major fields of research are the statistics of rigid body motion, with an emphasis on tectonic plate reconstructions, and the analysis of finite population sample surveys.



Correction to: Monitoring Coefficient of Variation Using CUSUM Control Charts

Phuong Hanh Tran, Huu Du Nguyen, Cédric Heuchenne, and Kim Phuc Tran

Correction to: Chapter 18 “Monitoring Coefficient of Variation Using CUSUM Control Charts” in: H. Pham (ed.), *Springer Handbook of Engineering Statistics*, https://doi.org/10.1007/978-1-4471-7503-2_18

The original version of this chapter was revised. The chapter was inadvertently published with author affiliation interchanged for the authors Dr. Huu Du Nguyen and Dr. Kim Phuc Tran.

The correction to this chapter is available at https://doi.org/10.1007/978-1-4471-7503-2_57

The updated original version of this chapter can be found at https://doi.org/10.1007/978-1-4471-7503-2_18

Index

- A**
- Aalen version of Cox model, 130
 - Absorbing process, 138, 142
 - Accelerated life tests (ALTs), 106, 117, 125, 423, 1008–1009, 1039
 - background and historical data, 117
 - illustration of test plans, 119–120
 - life distribution, 1028–1029
 - life-stress models, 1029–1031
 - lifetime model, 118–119
 - models, 423
 - plan design, 1035–1037
 - statistical analysis, 1031–1035
 - test plan development, 119
 - types of, 1027
 - Accelerates, 704
 - Acceleration factor (AF), 1008, 1009, 1015, 1018
 - Acceptable quality level (AQL), 223
 - Acceptance probability, 777
 - Acceptance rate, 777
 - Acceptance region, 36
 - Acceptance sampling, 221
 - Accumulated creep strain, 1010
 - Activation function, 630, 804, 913
 - Activity monitoring, 810–811
 - AdaBoost, 872
 - Adaptive banding, 890–892
 - Adaptive genetic algorithms, 638
 - Adaptive preventive maintenance, 531
 - Adaptive-weight Genetic Algorithm (AWGA), 640
 - Additive formation of multivariate utility functions, 211
 - Adjusted Rand index (ARI), 628
 - Advantage, 1080
 - Adversarial learning, 903, 909, 910, 912
 - Affine minorization, 680
 - Affinity propagation (AP), 808
 - Age replacement, 191, 543–552, 562–564
 - Agglomerative hierarchical methods, 626
 - Agglomerative method, 877
 - Aging, 54
 - phenomenon in reliability, 270
 - Aircraft spare parts, 280–284
 - Akaike information criterion (AIC), 51, 52, 71, 595, 782, 787, 881, 1048, 1049, 1084
 - Alhazmi Malaiya Logistic (AML) model, 468
 - α -mixing condition, 852
 - Alternating least squares (ALS), 819
 - Alternative prior distributions, 928
 - Amsterdam energy consumption, datasets for
 - cross-validation, 883
 - exploratory data analysis, 879
 - Keras, 883
 - LightGBM, 883
 - modelling selection criteria, 879–882
 - XGboosting, 883
 - Analysis
 - convex, 676
 - of market behaviour, 987
 - nonconvex, 676
 - Analysis of variance (ANOVA), 83, 251
 - Anderson-Darling test, 1016
 - Ansari-Bradley test, 362, 363, 370, 371, 373, 380
 - Ant colony optimization (ACO), 636
 - Aperiodic, 775
 - inspection planning, 526
 - Apollo, 1084
 - Approximate Bayesian computations (ABC), 581
 - Approximate confidence intervals, 1046, 1047
 - Approximate maximum likelihood estimation, 317–319
 - Approximate maximum likelihood predictors, 319–322
 - Approximation, stochastic, 687
 - Archimedean copulas, 937–939
 - ARIMA model, 203
 - Arrays, triangular, 695
 - Arrhenius model, 1029, 1035
 - Artificial intelligence, 982, 993
 - Artificial neural network, 629, 874
 - As-good-as-new repair, 190
 - Assembly line balancing (ALB), 657–659
 - Association, 938–941
 - analysis, 807
 - rule, 807
 - Asymptotic CD, 576
 - Asymptotic confidence intervals, 1045
 - Asymptotic relative efficiency, 1121
 - Asymptotic variance, 118, 122, 123
 - Attention deficit hyperactive disorder (ADHD) dataset, 827
 - Attribute control chart, 266
 - Attribute data, 262
 - Atypical observations, 605
 - Auto-regressive moving average (ARMA) model, 366
 - Autocorrelation, 114, 776
 - coefficient, 858
 - Automatic process control (APC), 203–204
 - Automatically controlled processes, 204–205
 - Availability, 425, 458
 - Average number of samples, 266
 - Average run length (ARL), 201, 335, 364
 - Average sample number (ASN), 223
 - Average total inspection (ATI), 223
 - Azzalini's skew normal (ASN) distribution, 87, 88

- B**
- Backpropagation, 630, 805, 866
 - Bagging, 631, 871
 - Ball grid array (BGA), 1005
 - Banded matrix estimation, 890
 - Bandwidth, 70
 - Banking risk, 989
 - Baseline cumulative hazard function, 1043
 - Baseline distribution, 45, 50, 51
 - Baseline hazard function, 832
 - Baseline reliability, 1043
 - Basquin law, 969
 - Bat-neural network multi-agent system, 983
 - Batch means estimator, 776
 - Bathtub curve, 417
 - Bath-tub failure rate, 55
 - Bayes estimates, 712, 767
 - Bayes factor, 585, 784–786, 788
 - Bayes posterior, 582
 - Bayes' rule, 4, 764
 - Bayes' theorem, 764–766
 - Bayesian analysis, 712–713, 781
 - Bayesian approach, 37, 530
 - Bayesian computation, 772–779
 - Bayesian cross validation, 783–784
 - Bayesian discrepant posterior phenomenon, 790
 - Bayesian estimator, 580
 - Bayesian hierarchical model, 771
 - Bayesian inference, 532
 - posterior propriety, conditions for, 925
 - posterior propriety, remedies for, 925
 - prior and posterior distributions, 924
 - Bayesian information criterion (BIC), 51, 52, 71, 628, 786, 892, 1048, 1084
 - Bayesian interval estimation, 769
 - Bayesian linear modelling, 770
 - Bayesian linear regression, 769–770
 - Bayesian marginalization paradox, 789
 - Bayesian method, 967, 1037
 - Bayesian methodology, 789
 - Bayesian models
 - averaging, 786
 - Bayesian computation, 772–779
 - Bayesian interval estimation, 769
 - Bayesian regression, 769–772
 - comparison, 929
 - conditional probability, 764
 - conjugate prior, 766
 - different fields of research, 787–789
 - empirical Bayes, 767–768
 - hierarchical Bayes, 767
 - model comparison, 782–787
 - monitoring convergence, 780–782
 - Monte-Hall problem, 765
 - multinomial distribution and Dirichlet prior, 766
 - point estimation, 768
 - reference prior, 766–767
 - types of prior, 765–766
 - Bayesian modeling, 211
 - Bayesian posterior, 970
 - Bayesian p -value, 781
 - Bayesian regression, 769–772
 - Bayesian shrinkage, 623
 - Bayesian smoothing, 787
 - Bayesian test planning, 106
 - Bayesian update, 530–533
 - Bayesian, fiducial, and frequentist (BFF), 580
 - Behavioral analysis, 987–988
 - agents, 996
 - Behavioral perspective, 988
 - Beliefs, obligations, intentions, and desires (BOID) architecture, 992
 - Belt structure, 241
 - Bernoulli distribution, 771
 - Bernoulli variable, 593
 - Bernstein inequality, 856
 - Bessel function, 146
 - Best linear equivariant estimators (BLEEs), 162
 - Best linear unbiased estimators (BLUEs), 710, 729
 - Best linear unbiased prediction (BLUP), 72, 78
 - Best permutation algorithm (BPA), 892
 - Beta-binomial model, 781
 - Beta-Birnbaum-Saunders distribution, 44
 - Beta-BS distribution, 51
 - Beta density, 19
 - Beta distribution, 11–12, 20, 484, 485
 - Beta exponential (BE) distribution, 394
 - Beta function, 43
 - Beta-G family, 43–44
 - Beta-log-logistic distribution, 44
 - β -mixing, 856, 858
 - coefficient is dimension-dependent, 859
 - condition, 856
 - Beta-Weibull distribution, 44
 - Bi-dependence, 130
 - Bias, 323
 - Bicriteria MXF/MCF model, 650–651
 - Bicriteria network design (BND), 650, 651
 - Bicriteria transportation model with fuzzy coefficients, 663
 - Big data, 789
 - Bike-sharing data, 69–73, 77–79, 82, 83
 - Bin-packing model, 644–645
 - Binomial choice model, 1077–1078
 - Binomial distribution, 11–12, 781
 - Binomial model extension based on total cost function
 - industrial application, 290–292
 - service level optimization, 289–290
 - simulative model, 290–291
 - Biostatisticians, 932
 - Birnbaum-Saunders (BS) distribution, 88
 - Bivariate generalized linear failure rate (BVGLF) distribution, 746
 - Bivariate exponential distribution, 189
 - Bivariate exponentiated Rayleigh (BVER) distribution, 746
 - Bivariate exponentiated Weibull (BVEW) distribution, 746
 - Bivariate gamma distribution, 271
 - Bivariate Gaussian random vector, 860
 - Bivariate GE-geometric distribution, 758–759
 - Bivariate generalized exponential (BVGE) distribution, 746
 - Bivariate Kumaraswamy distribution, 742
 - Bivariate normal correlation, 586–587
 - Bivariate normal distribution, 589
 - Bivariate PHM-geometric distribution, 756–758
 - Bivariate proportional reversed hazard distribution, 746
 - Bivariate replacement policy, 544
 - Bivariate Weibull-geometric distribution, 755–756
 - Black Belts, 241
 - Black-box testing strategy, 437
 - Block replacement, 191
 - Blockwise coordinate descent algorithm, 827
 - Blood glucose monitoring data, 379
 - Bonferroni correction, 623
 - Bootstrap, 108, 114, 836–839, 841, 940
 - confidence intervals, 60, 712, 1046

- Borel measurable function, 688
 - Bottom-up mechanism, 991
 - Boundary element method (BEM), 974
 - Bounds
 - linear upper, 683
 - proximal upper, 683
 - quadratic upper, 683
 - Box-Cox transformation, 271
 - Boxplot, 594
 - Breast cancer data analysis, 841
 - data processing and description, 841
 - fitting survival random forest, 841, 842
 - Brier score, 838, 839, 844
 - Brown-Proschan model, 516
 - Budget, 1050
 - Bug detection phenomenon, 471
 - Burn-in period, 55
 - Business planning system, 811
 - Buy-and-Hold strategy, 998
- C**
- Calibration factor, 482
 - Canadian weather station data, 68
 - CANDECOMP/PARAFAC decomposition, 819
 - Canonical correlation, 95
 - Canonical maximum likelihood (CML), 940
 - Canonical representation, 933
 - Capacitated plant location problem (cPLP), 668
 - Capture-recapture survey, 787
 - Cartesian product, 501
 - Catastrophic failure, 497, 499
 - Categorical variable, 594
 - Categorization scheme, 276
 - Cauchy distribution, 22
 - Cauchy probability density, 22
 - Cauchy-Schwarz Inequality, 6
 - Cayley-Klein parameters, 1071
 - c chart, 262, 265
 - cDNA, 632
 - Censored data, 25, 316, 327
 - Censoring, 107, 421, 832
 - hybrid, 422
 - interval, 422
 - progressive, 422
 - proportions, 323
 - random, 423
 - type I, 421
 - type II, 421
 - Censoring unbiased transformations (CUTs), 844
 - Central limit theorem (CLT), 15, 365, 773, 776, 970
 - Central moments, 48
 - Centralized network design, 648, 649
 - Chain sampling, 222
 - and deferred sentencing, 227
 - Chain-deferred (ChDP) plan, 228
 - Champions, 241
 - Change-point based control chart, 368, 369, 379
 - Change point detection (CPD), 362
 - Characteristic function, 88–90, 95
 - Characteristic life, 1014
 - Chebyshev's inequality, 903
 - Checked with caution, 858
 - Chernoff inequality, 903
 - Chi-square automatic interaction detection (CHAID), 802
 - Chi-square distribution, 586
 - Chi-square test, 30
 - Chi-square type statistic, 1047
 - Chi-squared, 595
 - Chip design, 1010
 - Choice-making behavior, 1075
 - Cholesky factor matrices, 889, 890, 892–894, 897
 - Cholesky-based matrix estimation, 888
 - Cholesky-based precision matrix estimation, 895
 - ChSP-1 plan, 223, 233
 - ChSP-4 plan, 223
 - ChSP-4A plan, 223
 - C-index, 838, 839, 842
 - Circuit pack assemblies (CPA), 308
 - Clarke subdifferential, 677, 678
 - Classic decision rule, 1098
 - Classic multivariate statistical methods, 932
 - Classical inference, 746–752
 - Classification, 804
 - trees, 622, 630
 - Classification and regression tree (CART), 596, 802, 870
 - Clayton copula, 938, 941
 - Climate and environmental change, 949, 961
 - Climate data, on Minneapolis, USA, 377
 - Cluster analysis, 807–809
 - Clustering, 625
 - analysis, 627–628
 - classification, 622
 - distance, 625–626
 - methods, 626–627
 - Codelets, 995
 - Coefficient of determination, 78
 - Coefficient of thermal expansion (CTE), 1005, 1007
 - Coefficient of variation (CV), 212, 334
 - Coffin-Manson equation, 1006
 - Coffin-Manson model, 1008, 1030
 - Cognitive agents, 984, 991–996
 - Cognitive architecture
 - cognitive agent characteristics, 991
 - definition, 989
 - emergent architecture, 991
 - examples of, 990
 - hybrid architecture, 991
 - symbolic architecture, 990–991
 - Cognitive cycle, 996
 - Cognitive distortions, 988
 - Cognitive technologies, 982
 - Cohen's progressive censoring, with fixed censoring times, 165
 - Coherent risk measures, 955
 - Coherent systems, 7, 8, 704, 706, 710, 712, 713, 729
 - Cold-standby system, 142
 - Combination warranty (CMW), 187
 - Combinatorial optimization problems, 642–646
 - Combined arrays, 207–208
 - Common cause, 361
 - failure, 62
 - Competing risks, 516–517
 - Competing risks models, for independent failure models, 1052–1053
 - Competitive to the copula methodology, 130
 - Complex systems, 419, 420
 - application to, 404
 - repairable reliability, 404
 - 3-Component, 713
 - Component events, 115
 - Compound Poisson process, 503–505
 - Concave bases (C-splines), 113
 - Concentration inequalities, 903–904

- Concordance function, 934
- Concordance measures, 934
- Conditional distribution, 6
- Conditional expectation, 689
- Conditional probability, 4, 138, 139, 764, 765
- Conditional reliability, 9, 14
- Conditional single sampling (CSS) plan, 229
- Condition-based maintenance, 496, 522
- Conference matrix, 305–308, 310
- Confidence, 807
 - bands, 73, 79, 83
 - curve, 576
 - density, 576
- Confidence distribution (CD), 576, 589
 - asymptotic CD, 576
 - CD-based inference, 578–579, 585–587
 - CD random variables, 586
 - definition, 575
 - fusion learning, 580
 - multivariate CDs, 580
 - upper and lower CDs, 577–578
- Confidence intervals (CIs), 27, 28, 102, 111, 576, 578, 579, 704, 729, 970
 - bootstrap CIs, 712
 - missing information principle, 711–712
 - observed Fisher information matrix, 711
 - reliability prediction, 489–490
 - β -RFE model, 489
 - γ -RFE model, 488–489
- Confidence limits, 33
- Confidentiality, 458
- Confounding factors, 822
- Conjugate prior, 532, 766, 772
- Connection weights, 629
- Consistency, 23, 132, 134
- Consistent Akaike Information Criterion (CAIC), 51, 52
- Constant failure rate, 141
- Constant-stress ALTs (CSALTs), 1040, 1043, 1050, 1052
 - multiple stress, 1036
 - plan design, 1036
 - single stress, 1035
- Context dependency, 1085
- Context-dependent model, 1080
- Contingency, 63
- Contingency table, 597
- Continuous-time predictive maintenance modeling
 - dynamic maintenance policy, framework of, 526–532
 - dynamic policy, 538–540
 - gradual deterioration and predictive maintenance, 523–526
 - long-term assessment, decision variables based on, 535–538
 - maintenance and inspection decision rules, 525–526, 533–534
 - maintenance cost, 535
 - stochastic processes, 523–525
 - unknown degradation parameters and update, 532–533
- Control charts, 333
- Convergence
 - finite, 680
 - of sample auto-covariance matrices, 862
- Convex analysis, 676
- Convex minorants, 682
- Convexity
 - strong, 677
- Convolutional neural network (CNN), 805, 874, 907
- Cook's distance, 614
- Cook's statistic, 609, 611, 614, 616, 617
- Coordinate measuring machine, 1110
- c -optimality, 121
- t Copula, 936
- Copula(s), 736, 933, 944
 - Archimedean copulas, 937–939
 - elliptical copulas, 935–937
 - method, 516–517
 - models, for correlated failure models, 1053–1054
- Correct model selection rate, 324
- Correlated failure modes, 1053
- Correlation, 594
 - coefficients, 135
 - matrix, 94
- Cost functions, 475, 476, 478
 - higher order time-dependent, 1097–1098
 - linear time-dependent, 1096–1097
- Cost of poor quality (COPQ), 241
- Cost rate, 535
- Cougaar, 992
- Counting processes, 145
- Coupling, 850
- Covariance function, 69–71
- Covariance matrix, 607, 612, 770, 856, 888, 896, 967, 977
 - estimation, 889–890
- Covariance stationary, 146
- Covariate-adjusted tensor classification in high dimensions, 825–827
- Covariation matrix, 951
- Coverage, 577, 578, 582, 583, 586, 589
 - probability, 1047
- Covering number, 907, 912–916
- Cox model, 833
- Cox proportional hazards regression model, 922, 924–927
- Cramer-von Mises-Smirnov type statistic, 1048
- Cramer-von-Mises test, 362, 363, 369–373, 380
- Cramér–Rao inequality, 23
- Crash tests, 598
- Credible set, 769
- Creep strain, 1007
 - density-based model, 1007
- Critical region, 36
- Critical-to-quality (CTQ), 243–244
- Cross-covariance, 78
- Cross-validation (CV), 70, 596, 783–784, 883
- Croston's method, 277
- Cumulative baseline hazard function, 923
- Cumulative count of conforming (CCC) chart, 263
 - CCC-r chart, 264
 - CUSUM control chart, 263
 - EWMA control chart, 263
 - geometric distribution, 263
- Cumulative damage models, 969–971
- Cumulative distribution function (CDF), 4, 15, 43–46, 51, 54, 88–90, 107, 115, 118, 264, 423, 424, 705, 706, 709, 933, 935, 936, 940, 1093
- Cumulative exposure model, 109, 1054
- Cumulative failure rate, 55
- Cumulative hazard function (CHF), 832, 837
- Cumulative number of failures, 108
- Cumulative number of faults, 492
- Cumulative quantity control (CQC) chart, 269–270
- Cumulative results criterion (CRC), 222, 224
- Cumulative shock damage, 512
- Cumulative-sum (CUSUM)
 - approach, 232
 - chain sampling and, 232–233
 - control chart CCC, 263

- structure, 271
- ZIP distribution, 267
- Curvic coupling, reliability and life prediction
 - probabilistic LCF life prediction, 975–977
 - stress and strain states, 974–976
 - time-dependent reliability, 977
- CUSUM chart, 201, 362, 366, 367, 379
- CUSUM control chart
 - accuracy error, 352
 - average run length, 337
 - average time to signal, 337
 - in-control state, 353
 - extensive overview of, 334
 - FIR strategy, 354
 - Fredholm integral equation method, 339
 - implementation of, 337
 - Markov chain, 338
 - measurement error, 333, 344
 - methods to calculate, 338
 - monitoring simple linear profiles, 334
 - one-sided control chart, 340
 - optimal control chart coefficients, 345
 - out-of-control state, 354
 - performance of, 333, 341
 - precision error ratio, 352
 - sequence works, 336
 - sintering process, 344
 - SPC literature, 336, 357
 - standard-deviation, 338
 - statistics, 336
 - two one-sided, 340
 - upward, 342
- Cut vector, 8
- D**
- Damage probability, 12
- Damage tolerance, 966, 971–974
- Daniell-Kolmogorov consistency theorem, 132
- D* approach, 323
- Data, 709
 - analysis, 752–753
 - regularization, 951, 954
 - science, 216
 - subanalytic, 681
 - warehouse, 800
- Database, 798
- Data freeze date (DFD), 107
- Data mining, 797, 817
 - activity monitoring, 810–811
 - applications, 798
 - artificial neural networks and convolutional neural networks, 804–806
 - classification problems, 806
 - flow chart, 799
 - implementation of, 814
 - improvements, manufacturing, 798
 - linear and additive methods, 801–802
 - Mahalanobis–Taguchi System, 811
 - manufacturing process modeling, 811–812
 - object detection, 813
 - prerequisites for, 798
 - process, 800
 - public health surveillance, 813
 - research and applications, 810–813
 - software, 809–810
 - success of, 813
 - supervised learning, 801–806
 - support vector machines, 805–806
 - trees and related methods, 802–804
 - unsupervised learning, 806–809
- DC components, 678
- DC criticality, 678, 679
- DC decomposition, 678, 684
 - successive, 696
- DC duality transportation, 680
- DC program, 681
- DC programming and DCA
 - deterministic, 676–684
 - stochastic, 687–696
- DCA, 677, 679
 - and convergence properties, 680–681
 - developments and open problems, 682
 - online stochastic, 688–696
 - standard convex/nonconvex programming approaches, 682–684
 - with subanalytic data, 681–682
- dChip programs, 625
- Debugging process, 469
- Decision making approach, 957–961
- Decision rule, 525, 535, 1098
- Decision support systems, 981
- Decision tree, 603
 - method, 870
- Decision variables, 535
- Decision-making, 987, 1094
- Deep convolutional neural network (DCNN), 449
- Deep feedforward neural network (DFNN), 449
- Deep learning, 868, 983, 984
 - agents, 996
 - approaches, 813
- Deep neural networks (DNNs), 902, 907–910, 912, 913, 916
- Deep text analysis, 984
- Default risk, 989
- Defect identification process, 459
- Defect prevention, 438
- Defects per million opportunities (DPMO), 240, 243, 254
- Deferred-dependent plans, 230
- Define, Measure, Analyze, Design and Verify (DMADV), 246–248
- Define, Measure, Analyze, Improve and Control (DMAIC), 242
 - opportunities, 256–257
 - process, 242–244
 - six sigma road map, 247
 - tools to support the process, 244–247
- Degenerate, 857
- Degradation analysis, 942–943
- Degradation data analysis
 - background and data, 112, 113
 - model for covariates, 114
 - model for degradation paths and parameter estimation, 113, 114
 - reliability prediction, 114, 115
- Degradation modeling, 56, 522
- Degradation path, 114
- Degradation process, 532
- Degradation rate, 112
- Degrees of freedom, 594
- Delta method, 1045
- Demand spare parts
 - aircraft, 280–284
 - binomial model extension based on total cost function, 289–292
 - forecast problem of, 276–279
 - forecasting methods applicable to, 280–281
 - models based on binomial distribution, 286–289

- Demand spare parts (*cont.*)
 - operational management of, 301–302
 - packaging machines, 287–289
 - Poisson models, 284–286
 - standard vs. custom-made components, 295, 302
 - Weibull extension, 292–294
 - Dense design, 69
 - Density-based clustering (DBSCAN), 808
 - Density-based spatial clustering of applications with noise (DBSCAN), 877
 - Density function, 70, 1093
 - of log normal distribution, 417
 - Density power divergence (DPD) measure, 1051
 - Dependence adjusted norm (DAN), 854, 855
 - Dependence parameter, 1053
 - Dependent-deferred sentencing, 222
 - Dependent deferred state (DDS) plan, 228
 - Dependent random matrices, 856
 - Dependent vulnerabilities, 460
 - Depth-first search (DFS), 649
 - Descriptive statistics, 1009
 - Descriptive theory, 987
 - Design engineer, 1015
 - Design failure modes and effects analysis (DFMEA), 250
 - Design for Six Sigma (DFSS), 244, 246–247
 - Design of Experiments (DOE), 243, 245–246
 - Desirability, 209
 - Deterioration, 522, 523, 525, 527, 531, 535
 - Deterministic DC programming and DCA, 676–677
 - basic results, 678–682
 - preliminaries, 677–678
 - standard convex/nonconvex programming approaches, 682–684
 - Deterministic forests, 631
 - Deterministic model, 966
 - Deviance information criterion (DIC), 782, 926, 927, 929
 - Diagonal matrix, 913
 - Dimension folding, 821
 - Dimension reduction, 821
 - techniques, 878
 - Dirichlet distribution, 766
 - Dirichlet prior, 766
 - Discounted warranty cost (DWC), 191, 193
 - Discrepant posterior phenomenon (DPP), 790
 - Discrete choice modeling, 1076–1077
 - RAM, 1080–1081
 - RRM, 1079–1080
 - RUM, 1077–1079
 - Discrete continuous processes, 500
 - Discrete distribution, 13
 - Discrete Fourier transformation, 111
 - Discretization theorem, 907
 - Discriminant analysis, 84
 - Discriminate, 1042
 - Discriminative classification model, 868
 - Dispersion matrix, 935
 - Distance-based test statistic, 1048, 1052
 - Distribution centers, 63
 - Distribution estimator, 576
 - Distribution of the remaining life (DRL), 110
 - Distributional change, 362
 - Divisive hierarchical methods, 626
 - Divisive method, 877
 - DNA, 622
 - Domino effect, 63
 - D-optimal design, 208
 - D-optimality, 121, 124
 - DSD-augmented designs (ADSDs), 306
 - d-stationarity, 679
 - Dual objectives, 121, 123
 - Dudley's theorem, 907
 - Durbin-Watson test, 377, 379
 - Dynamic covariates, 106, 109, 111–116, 125
 - Dynamic decision-making, 540
 - Dynamic maintenance, 522
 - Dynamic maintenance policy, 526–527
 - grouping and opportunistic maintenance, 527–530
 - uncertainty in degradation models, 530–532
 - Dynamic policy, 532–540
 - Dynamic system, 1090
- E**
- Earth sciences, 959
 - Economic approach, 284
 - Economic dependency, 527
 - Economic design method, 265
 - Effective sample size, 776, 780
 - Effort-dependent vulnerability discovery model, 461
 - Eigenanalysis, 616
 - Eigenfunctions, 69, 71
 - Eigenvalues, 856
 - Elastic net, 876
 - penalty, 821
 - Electroencephalography (EEG), 817
 - Electronic packaging
 - accelerated life testing, 1008–1009
 - background and research motivation, 1005–1007
 - cause of failure, 1007
 - fatigue life prediction, 1008
 - finite element method, 1007
 - issues, 1004
 - modeling and verification, 1010–1015
 - parameter uncertainty, 1009
 - probabilistic design and reliability, 1009–1010
 - Electronic product, 1019
 - Elliptical copulas, 935–937
 - EM algorithm, 1043–1045, 1054
 - Emergent architecture, 991
 - Emotion(s), 988
 - of regret, 1079
 - Empirical analysis
 - data, 466
 - parameter estimation and goodness of fit criteria, 465
 - validation result of generalized coverage dependent, 468–469
 - validation results of user dependent and effort dependent VDMs, 466–468
 - Empirical Bayes, 767–768
 - Empirical Bayesian method, 624
 - Empirical distribution, 624
 - Empirical likelihood, 59
 - Empirical loss, 676
 - Empirical Rademacher complexity, 904
 - Empirical risk minimizer (ERM), 901
 - Engineering based reliability test (ERT), 1025
 - Entropy, 631
 - Envelope, 822, 825
 - tensor response regression, 821–823
 - Environmental condition, 1004, 1005
 - Environmental factor, 491
 - Environmental risk analysis, 960
 - Equicontinuity, 692
 - uniform, 692

- Equivalent business days (EBD), 811
 - Ergodic process, 138, 143
 - Erratic demand, 276
 - Error, 435
 - Estimated standard error, 1045
 - Estimations, 530
 - Euclidean norm, 858
 - Euler-Mascheroni constant, 710
 - Evaluation function, 998
 - Evolutionary algorithms (EAs), 636
 - Evolutionary programming (EP), 636
 - Evolutionary strategy, 983
 - Evolution strategy (ES), 636
 - EWMA chart-type charts, charts based on change point detection (CPD), 362
 - Exact, 586
 - maximum likelihood, 939
 - penalty, 682
 - Excess risk, 902, 904
 - Exchange rate risk, 989
 - Exhaustive search algorithms for, 803
 - Expectation, conditional, 689
 - Expectation maximization (EM) algorithm, 89, 90, 95, 97–102, 684, 704, 705, 748–752, 759, 773, 790
 - Expected acceptance rate, 777
 - Expected average run length (EARL), 335, 341
 - Expected cycle length, 508–509
 - Expected discounted warranty cost (EDWC), 190
 - Expected loss, 676
 - Expected maximum utility, 1082
 - Expected profit, 193
 - Expected value prediction method, 320
 - Expected warranty cost (EWC), 190
 - Explanatory methods, 279
 - Exponential distribution, 14, 34, 138, 269–270, 417, 737, 1028, 1041, 1044, 1051, 1054, 1055
 - Exponential family, 820
 - Exponential models, 423, 1041, 1054–1055
 - Exponential random variable, 150
 - Exponential smoothing, 277
 - Exponentially weighted moving average (EWMA) chart, 201, 263, 367, 379
 - ZIP distribution, 267
 - Exponentially weighted moving average (EWMA) control chart, 334
 - Exponentiated distributions, 47
 - Exponentiated Log distribution, 44
 - Expression profiles, 627
 - Extreme-value distribution, 21
 - Eyring model, 1030, 1041
- F**
- Face detection, 813
 - Factor analysis methods, 812
 - Factorization pattern, 130
 - Fail-safe, 965
 - Failure, 522
 - censoring, 1027
 - defect data, 272
 - definition, 435
 - intensity, 482
 - mechanism, 190
 - mode, 242, 1052
 - process, 148
 - time, 110, 270, 940, 944
 - Failure-detection rate, 487
 - Failure modes and effects analysis (FMEA), 243–245
 - Failure rate, 10, 55, 544, 546, 565, 968, 1017
 - exponential distribution, 417
 - function, 416
 - False-alarm probability, 262–264, 270
 - False discovery rate, 623
 - False negative (FN), 896
 - False positive (FP), 896
 - False selection loss (FSL), 896
 - Fast algorithm for classification trees, 803
 - Fast gradient sign method (FGSM), 910, 911
 - Fast initial response (FIR) strategy, 335, 354
 - Fatigue, 1007
 - failure, 1007
 - life, 1004, 1008
 - test, 117, 122
 - Fatou's lemma, 688
 - Fault
 - debugging, 471
 - definition, 435
 - detection phenomenon, 470
 - diagnosis, 206
 - seeding and input domain based models, 433
 - Fault count (FC) models, 431
 - Fault-detection rate, 482
 - F distribution, 17
 - Features, 629
 - Feedback control scheme, 203
 - Feedforward neural network, 874
 - Fiducial distribution, 575
 - Fiducial factor, 585
 - Fiducial inference, 581–585
 - Field condition, 1005
 - Field environments, 485
 - Field operations, 486
 - Financial decisions, 981
 - behavioural analysis, 987–988
 - fundamental analysis, 984–986
 - risk, 988–989
 - technical analysis, 986–989
 - Financial markets, 996
 - Financial risk, 989
 - Finite convergence, 680
 - Finite element method (FEM), 974, 1007
 - Fintech, 982, 983
 - First-order D-efficiency, 308
 - First-order derivatives, 707
 - First passage time, 523
 - Fishbone diagram, 245
 - Fisher information, 58
 - in hybrid censoring schemes, 169
 - Fisher information matrix, 27, 89, 98, 100, 489, 735, 752, 753, 939, 1048
 - Fisher-von Mises distribution, 1061
 - Fisher–von Mises–Langevin distribution, 1123
 - Fitting distributions, 51
 - Fitting OLR models, 594–595
 - Fixed, 858
 - Flat earth approximations, 1068
 - Flexibility, 439
 - Flexible job-shop scheduling problem (FJSP), 654–656
 - Flexible regression models, 806
 - Flow-shop sequencing model, 652–653
 - Foldover designs (FODs), 306
 - Forecast-based monitoring methods, 202, 204

- Forecasting methods, 279
 - characterization, 279–280
 - classification of, 278
 - explanatory methods, 279
 - judgmental method, 279
 - monitoring methods, 279
 - quantitative methods, 279
 - related to spare parts demand, 280–281
 - technological methods, 279
 - FOREX, 997
 - Forward adaptive banding, 891–892
 - Fourier transform, 856
 - Fourier transform infrared (FTIR), 112
 - Fréchet-Hoeffding bounds, 934
 - Fréchet subdifferential, 677
 - Fraction of variance explained (FVE), 71, 77
 - Fractional factorial designs (FFDs), 305
 - Frank copula, 938, 1053
 - Fredholm integral equation, 338
 - Free repair warranty (FRPW), 187
 - Free replacement warranty (FRW), 187
 - Frobenius norm, 820, 908
 - Fubini theorem, 697
 - Full-service warranty (FSW), 187
 - Function-on-function regression, 82
 - Functional classification, 84
 - Functional data analysis (FDA), 68, 69, 82–84
 - FLR, 77–79
 - FPCA, 69–73
 - software packages, 84
 - Functional dependence measure, 851, 853–855
 - Functional dependence setting, 853
 - Functional linear regression (FLR), 69
 - estimation of regression function and R^2 , 78
 - implementation details, 79
 - model for, 77–78
 - prediction and confidence bands, 78–79
 - Functional linear regression model, 77
 - Functional principal components analysis (FPCA), 69
 - implementation details, 73, 73
 - mean function, eigenvalues and eigenfunctions, 70–71
 - model for, 69–70
 - PC scores, 71, 72
 - prediction of trajectories and confidence bands, 72–73
 - Functional regression, 78
 - Fundamental analysis, 984–986
 - Fundamental analysis agents, 996
 - Fusion learning, 575, 580
 - Fuzzy logic controller (FLC), 638
- G**
- Game theory, 64
 - Gamma distribution, 18, 418, 484, 492, 1042, 1047, 1051
 - Gamma function, 1010
 - Gamma-G family, 45
 - Gamma models, 1042
 - Gamma priors, 767
 - Gamma process, 524–525
 - Gauge R&R, 243, 245
 - Gauss-Newton, 89
 - Gaussian conjugate model, 790
 - Gaussian distribution, 903, 973
 - Gaussian likelihood, 970, 971
 - Gaussian probability distribution, 967
 - Gaussian random variable, 904
 - Gaussian random vector, 860
 - Gaussian sequences, 859
 - Gaussian variable, 966, 967
 - GE-geometric (GEG) distribution, 758
 - Gelfand-Dey estimator, 786
 - Gelman-Rubin diagnostic, 780–781
 - Gene filtration, 622
 - General additive model, 113
 - General framework of systems health monitoring and management, 216
 - Generalization bound, 902
 - Generalization error, 902, 907, 908, 910, 911
 - Generalized additive model (GAM), 802
 - Generalized coverage-dependent model, 463
 - Generalized distributions, 51
 - Generalized exponential distribution, 737
 - Generalized fiducial distribution (GFD), 581–583
 - for discrete distributions, 584
 - in irregular models, 583
 - model selection, 584
 - nonparametric fiducial inference with right censored data, 583–585
 - nonparametric GFD based inference, 587
 - probability measure, 581
 - Generalized fiducial inference, 576
 - Generalized gamma distributions, 1042, 1049
 - Generalized gamma models, 1042–1043
 - Generalized inference approach, 586
 - Generalized likelihood ratio test, 202
 - Generalized linear model (GLM), 210, 771–772, 802, 820, 821
 - Generalized Poisson distribution, 268, 269
 - Generalized statistical distributions
 - beta-G family, 43–44
 - fitting distributions, 51
 - gamma-G family, 45
 - Kumaraswamy-G family, 44–45
 - linear representations, 46–47
 - Marshall and Olkin-G family, 46
 - mathematical properties, 47–49
 - real data illustrations, 51–52
 - T-X family, 49–50
 - Generalized weighted indecisive voting, 1098
 - Generally weighted moving average (GWMA) technique, 335
 - General path models, 106
 - General probability space, 850
 - Generating function, 49
 - Generative adversarial networks (GANs), 912
 - Generative classification model, 868
 - Generative models, 912
 - Genetic algorithms (GAs), 636
 - adaptive, 638
 - basic, 636–637
 - combinatorial optimization problems, 642–646
 - FLC, 638
 - general structure, 637
 - hybrid, 637–638
 - with learning, 641
 - location and allocation problems, 667–671
 - logistic network problems, 662–667
 - multiobjective optimization problems, 639–641
 - network design problems, 646–651
 - reliability design problem, 659–662
 - scheduling problem, 652–659
 - Genetic programming (GP), 636
 - Genomic data, 631
 - Geo-referred filter, 954
 - Geo-space, 952

- Geometrically α -mixing, 858
 Geometrically ϕ -mixing, 857
 Geometrically decaying τ -mixing, 857
 Geometrically stable, 90
 Geometric distribution, 13, 263
 Geometric ergodicity, 776
 Geometric random variable, 88
 Geometric skew normal (GSN) distribution, 88–94
 Geometric statistical problems, 1070
 Germline mutations, 630
 Geweke diagnostic, 780
 Gibbs algorithm, 773
 Gibbs sampler, 775, 778–779
 Gini index, 834
 Global workspace, 997
 Godambe information matrix, 940
 Goel and Okumoto imperfect debugging model, 430
 Goel-Okumoto Non-homogeneous Poisson process model, 432
 Goldmann model, 1008
 Goldstein–Levitin–Polyak projection algorithm, 683
 Goodness of fit, 30, 450, 784
 Google chrome, 466
 Google flu trend (GFT), 813
 Gradient boosting algorithm, 872
 Gradient vector, 98
 Graphical Evaluation and Review Technique (GERT), 223
 Graphical Lasso (Glasso) model, 888
 Graphical user interfaces (GUIs), 435
 Grassmann manifolds, 823
 Green belts, 241
 Grouping strategy, 528
 Groups of outliers, 610
 eigenvectors of influence and sensitivity matrix, 611–613
 initial clean set, methods based on, 610–611
 Growth curves, 68
 Growth function, 905
 Γ -Gumbel, 45
 Gumbel copula, 938
 Gumbel distribution, 45, 1077, 1080
 Gumbel-Hougaard copula, 1053, 1054
 GZIP-CUSUM chart, 269
- H**
- Hadamard matrix, 306, 308
 Hadamard-matrix based MLFODs, 306
 Harris recurrent, 535
 Hastings ratio, 777, 778
 Hausdorff space, 687
 Hawkins's approximation, 338
 Hazard function, 10, 832, 833, 838, 843, 844
 gamma distribution, 418
 Hazard rate function (HRF), 44, 46, 416
 Hazard rates, 1041, 1042
 Heavy tailed, 88–90, 94, 95
 Helicopter's main rotor blade, 27
 Hellinger criterion, 1067
 Hessian matrices, 773
 Heterogeneity, 531
 Heterogenous MTRP, 116
 Heuristic choice protocols, 1085
 Heuristic distortions, 988
 Hierarchical Bayes, 767
 estimate, 788
 Hierarchical Bayesian linear regression, 770
 Hierarchical clustering, 628, 808, 877
 Hierarchical functional data, 83
 Hierarchical linear modelling, 770
 High dimension, 858
 dataset, 818
 statistical theory, 851
 Higher moments, 90
 Highest posterior density (HPD), 769, 926
 High leverage observations, 612
 Highly accelerated life test (HALT), 1026
 Hilbert space, 69, 953
 Hilbert-Schmidt norm, 861
 Hoeffding decomposition, 858
 Hoeffding's inequality, 860, 904
 Holder inequality, 6, 689
 Holt Winter's models, 277
 Hotelling's T^2 statistic, 206, 940
 Housing Boston data, 616–617
 Huber estimate, 1113
 Human behavior modeling, 1075
 Hybrid architecture, 991
 Hybrid genetic algorithms, 637–638
 Hybridization, 622
 Hyper-parameters, 532, 712
 Hypergeometric distribution, 13
 Hyperparameter, 768, 788
 Hypothesis testing, 578, 579
- I**
- IBM binomial and Poisson models, 432
 Image registration, 1061, 1111
 diagnostics, 1124–1127
 least squares estimation, 1113–1115
 location problems and regression problems, 1127–1128
 orthogonal and orthogonal matrices, 1111–1112
 Procrustes and spherical regression models, 1112
 statistical properties of M -estimates, 1118, 1121
 Imitators, 461
 Imperfect maintenance, 544, 546
 Imperfect repair, 187, 544
 Importance function, 774
 Importance sampling, 774
 estimator, 785–786
 Improper prior, 765, 766
 Impurity, 631
 Imputation, 599–601
 In-control process measurement, 361
 Incident, 435
 Incomplete gamma function, 45, 49
 Incomplete moment, 48
 Indecisive, 1091
 Indecisiveness, 1103
 2-Independence, 135
 Independence of irrelevant alternatives (IIA), 1081
 Independence sampler, 778
 Independent, 7, 820
 failure models, 1052–1053
 increments, 145
 random variables, 150
 Independent and identically distributed (i.i.d.) processes, 201
 Indifference price, 192
 Individual preference learning, 1085
 Individual preferences, 1075
 Indo-Australian plate, 1069

Industrial data mining, 810
 Inequality, 850, 852, 856, 915
 Infant mortality, 55
 period, 417
 Inference functions for margins (IFM), 939
 Inferential statistics, 251
 Infinite-dimensional, 68
 Infinitely divisible, 90
 Inflation risk, 989
 Influence function, 1124, 1125
 Influence matrix, 606, 611, 612, 616
 Influential observations, 606, 608–609
 Influential subsets, 606
 Information extraction, 984
 Information matrix, 1045, 1055
 Information retrieval, 984
 Informative prior, 790, 925
 Innovators, 461
 Input, 1092
 Inspection planning, 525
 Instantaneous fault-detection rate, 447
 Integral equation/inequality, 130
 Integrity, 458
 Intensity function, 116, 149
 Interaction, 594
 Interactive Adaptive-weight Genetic Algorithm (i-AWGA), 641
 Interest rate risk, 989
 Intermittent demand, 276, 277
 International Software Testing Qualification Board (ISTQB), 435
 Interval availability, 143
 Interval censoring, 57
 Interval estimation, 33
 Interval parameter, 33
 Inverse-Gaussian process, 525
 Investors' behaviour, 987
 Irrational behavior, 1085
 Irreducible, 777
 ϕ -Irreducible Markov chain, 775
 ISO 9001 standard, 426
 Iterative algorithm, 821, 823
 Iterative dichotomize 3, 870
 Iterative methods, 97
 Iterative shrinkage-thresholding algorithm, 684
i-th equivalence class, 502

J

Jackknifed confidence intervals, 1045
 Jacobian determinant, 517
 Jacobian function, 582
 Jacobian matrix, 967
 Jeffrey's prior, 766, 767
 Jeffreys-type prior, 924–926, 928
 based on the first risk set, 928
 Jensen's inequality, 6
 Job-shop scheduling model, 653–654
 Joh Weibull (JW) model, 458
 Joiners as dependence functions, 130
 Joint cumulative distribution function (JCDF), 737
 Joint Electron Device Engineering Council (JEDEC), 1007
 Joint monitoring, 271
 Joint posterior distribution, 712
 Joint probability density function (JPDF), 737
 Joint survival function, 738, 754
 Judgmental methods, 279

K

Kalman amplification, 955
 Kaplan-Meier curves, 587
 Kaplan-Meier estimator, 59–60, 832, 834, 836, 837, 839, 842
 Karhunen-Loève theorem, 70, 71, 78
k-dependence, 131
 Kendall's tau, 934, 935, 938, 1053
 Keras, 883
 Kernel copulas, 956, 957
 Kernel density estimator (KDE), 970
 Kernel function, 70, 806
 Kernel principal component analysis (KPCA), 951, 954, 961
 Khatri-Rao product, 818, 821
k-independent, 131
 KJ method, 248–249
k-means, 622, 808, 809
 clustering, 877
 Knapsack problem, 643
K-nearest neighbours, 868
k-NN classifier, 806
 Knowledge discovery in databases, 797
 data preparation, 800
 databases and data warehousing, 799–800
 handling data, 799–800
 process, 798
 Kolmogorov type statistic, 1048
 Kolmogorov-Smirnov test, 31, 369
K-out-of-*n* structure, 420
k out of *n* system, 7, 61
 Kronecker product, 818
 Kullback-Leibler divergence, 1050
 Kullback-Leibler information, 941
 Kullback-Leibler loss, 893
 Kumaraswamy-G family, 44–45
 Kurtosis coefficient, 5
k-variate survival function, 130
 Kw-exponential, 45
 Kw-Weibull, 45

L

Lack-of-fit criterion, 801
 Lag-one autocovariance matrices, 861
 Lag-one vector autoregressive model, 861
 Lagrangian dual-objective function, 806
 Laplace approximation, 772–773, 785
 Laplace method, 968
 Laplace transform, 140, 144, 151, 484, 938
 Large-sample property on approximating *A*, 862
 Large-sum problem, 685
 Largest extreme value distribution, 418
 Lasso penalty, 891
 function, 615
 Latent variable models, 912
 Lead-free solder, 1007
 Learning, 990, 991
 Learning intelligent distribution agent (LIDA), 994
 Learning theory
 for deep learning, 907–909
 for robust adversarial learning, 909–911
 Least median of squares (LMS), 614
 Least squares (LS) criterion, 607
 Least squares estimation, 32, 107, 861
 group properties, 1113
 for hands data, 1115
 parameterization, 1115–1118

- in Procrustes model, 1114–1115
 - singular value decomposition, 1114
 - spherical regression model, 1115
 - Least squares linear regression, 875
 - Leave-one-out cross validation (LOO-CV), 782, 783
 - LED packaging process, 268
 - Ledoux-Talagrand contraction inequality, 905
 - Left-censoring, 57, 1040
 - Lepage test, 372
 - Leverage, 607, 609, 610, 612, 614
 - Liander, 879
 - Life distribution, 1019
 - exponential distribution, 1028
 - lognormal distribution, 1028
 - Weibull distribution, 1028
 - Life prediction, 1008
 - Life-stress models
 - exponential model, 1029
 - power model, 1030
 - Life testing, 706
 - Lifetime data, 55, 56, 58, 106–108, 119, 122
 - LightGBM, 883
 - Likelihood, 108
 - function, 26, 27, 486, 593, 923, 1041
 - principle, 764
 - Likelihood inference, 1043
 - confidence intervals, 711–712, 1045–1046
 - designs of CSALTs, 1050
 - direct maximization of observed likelihood, 706–709
 - EM algorithm, 1043–1045
 - initial values to algorithm, 710–711
 - model mis-specification analysis, 1046–1047
 - model selection, 1047–1049
 - St-EM algorithm, 709–710
 - test of proportional hazard rates, 1050
 - Likelihood ratio (LR) method, 58–59
 - Likelihood ratio test (LRT), 98, 1049
 - Lindley's approximation, 712
 - Linear and additive methods, 801–802
 - Linear combination, 47
 - Linear damage rule, 970
 - Linear discriminant analysis (LDA), 629, 802, 825, 826, 869, 887, 895
 - Linear function, 943
 - Linearized models, 969
 - Linear mixed effect model, 110, 114
 - Linear model, 822
 - Linear process, 852, 855
 - Linear regression, 32, 594, 975
 - model, 769
 - Linear representations, 46–47
 - Linear subspace, 822
 - Linear transportation model, 662–663
 - Linear upper bounds, 683
 - Link(s), 1107
 - function, 771
 - Lipschitz constant, 908, 911
 - Lipschitz continuous functions, 855
 - 1-Lipschitz functions, 851
 - Lipschitz function, 916
 - Liquidity risk, 989
 - Littlewood-Verrall Bayesian model, 431
 - Load-share system, 62
 - Local c -optimality design, 120
 - Local linear, 70
 - Local minimizer, 679
 - Local polynomial, 69, 70, 78, 84
 - Location–allocation problem (LAP), 667–668
 - capacitated plant location model, 668
 - with multi-facility service, 670–671
 - obstacle location–allocation model, 669–670
 - Location-scale changes, 374
 - Location-scale distribution, 316
 - Location-shift SGSN (LS-SGSN) distribution
 - maximum likelihood estimators, 97–98
 - model description, 95–97
 - simulation results, 100
 - testing of hypothesis, 98
 - Logarithm of the conditional moment, 1093
 - Log-BS (LBS) distribution, 89
 - Log-concave property, 388, 393
 - Log-F prior, 929
 - Logistic distribution, 418
 - Logistic function, 1094
 - Logistic-G family, 50
 - Logistic network problems, 662–667
 - Logistic regression, 771, 801, 820, 870, 911–912
 - Logistic regression tree (LRT), 596–598
 - Logistic vulnerability discovery model, 458
 - Logistics systems, 63
 - Logit, 593, 1046
 - model, 1077, 1081
 - Log-likelihood, 684, 1084
 - function, 97, 99, 707, 751, 821, 1078
 - ratio test, 728
 - Log-location-scale, 107
 - distribution, 118
 - Loglog distribution, 20, 27
 - Log-logistic distribution, 418
 - Lognormal, 107
 - density, 16
 - distribution, 16, 147, 1028
 - Log odds, 1078
 - Log partial likelihood function, 923, 926
 - Log-rank test, 834–835, 843
 - Łojasiewicz exponent, 681
 - Łojasiewicz subgradient inequality, 681
 - Long-range dependence, 852
 - Long-run, 535
 - cost rate, 528, 537, 538
 - Long-term, 529
 - trend, 114
 - Longitudinal data, 82
 - Loss
 - empirical, 676
 - expected, 676
 - function, 784, 913
 - Low turnaround index (LTI) parts, 284
 - Low-cycle fatigue (LCF), 975–977
 - Lower and upper CDs, 577, 578
 - Lower bound on $\beta(\sigma(X_0), \sigma(X_m))$, 860
 - Lower triangular matrix, 888
 - Lower-bound truncated exponential distribution, 15
 - Lumpy demand, 276
- M**
- Machine learning, 596, 641, 983, 984
 - AdaBoost, 872
 - algorithms based on applications, 868
 - approaches, 813
 - artificial neural network, 874
 - bagging, 871

- Machine learning (*cont.*)
- bias-variance trade-off, 868
 - CART, 870
 - classification algorithm, 868
 - deep learning, 868
 - elastic net, 876
 - goals, 866
 - gradient boosting algorithm, 872
 - gradient boosting machine, 873
 - history, 866
 - importance of, 866–867
 - K-nearest neighbours, 869
 - LASSO, 875
 - least squares linear regression, 875
 - linear discriminant analysis, 869
 - logistic regression, 870
 - Naïve Bayes algorithm, 870
 - parametric vs. non-parametric learning, 867
 - quadratic discriminant analysis, 870
 - random forest, 871–872
 - reinforcement learning, 867
 - ridge regression, 875
 - semi-supervised learning, 867
 - supervised learning, 867
 - support vector machines, 869
 - support vector regression, 876
 - unsupervised learning, 867
 - XGBoost, 873
- Magnetic resonance image (MRI), 817
- Magnitude-events, 271
- Mahalanobis–Taguchi System, 811
- Maintainability, 425
- Maintenance, 521
- cost, 535
 - engineering, 543, 544
 - groupings, 530
 - model, 517, 735
 - modeling, 522
 - planning, 522
 - policy, 496
 - strategies, 188
- Maintenance cost analysis, 512
- calculation, 512
 - expected cycle length, 513
- Majorization-minimization (MM), 682
- Malenad Tiger Landscape, 788
- Mann-Whitney test, 372
- Manson–Coffin law, 969
- Manson–Coffin low-cycle fatigue life model, 969
- Manson-Coffin model, 976
- Manufacturers, 186
- Manufacturing process control, 812
- Manufacturing process modeling, 811–812
- MAPLE, 472, 476
- Mapping function, 501
- Marginal α -mixing coefficient, 860
- Marginal distribution, 768, 932, 939
- Marginal factor, 132
- Marginal fiducial distribution, 584
- Marginal generalized fiducial probability of model, 585
- Marginal likelihood, 784, 785
- Gelfand-Dey estimator, 786
- Marginal posterior density, 770
- Marginal probability, 764
- measure, 857
- Marginal regret, 1079
- Marginalization paradox, 789
- Marginally consistent, 937
- Marine magnetic anomaly lineations, 1066
- Market behaviour, 986
- Markov chain, 535, 685, 775, 776, 778, 780, 851, 861
- approach, 225, 338
 - method, 338
 - and stationarity, 775–777
- Markov chain Monte Carlo (MCMC), 116, 122, 764, 774–779, 783–785, 789, 790, 926, 967
- Markov model, 496
- Markov modeling, 141, 144
- Markov process, 138, 424, 858
- characteristics, 138
 - mean time failures, 142–144
 - types based on reliability/availability, 138
- Markov property, 775
- Markov's inequality, 903
- Markovian approach, 191
- Marshall and Olkin-G family, 46
- Marshall-Olkin bivariate exponential (MOBE) distribution, 734, 740–742, 747, 757
- Marshall-Olkin bivariate Weibull distribution, 742
- Martingale theory, 689
- Masking, 606, 610, 612, 616
- Massart's finite lemma, 906
- Master Black Belts, 241
- Mathematical induction, 132
- Mathematical maintenance cost, 496
- Mathematical properties, 47–49
- Mathematical software packages, 1114
- Matlab, 70, 73, 79, 810
- Matrix exponential map, 1116
- Matrix normal distribution, 819
- Maximization approach, 739–740
- Maximum a posteriori, 773, 790
- Maximum likelihood (ML), 594, 939, 940, 943
- estimates, 23–25, 51, 52, 58, 98, 99, 101, 102, 119, 120, 486–487, 704, 705, 707, 709–713, 729, 735, 748, 751–753, 768, 820, 822–824, 826, 924, 1031, 1032, 1035, 1043–1046, 1054, 1055, 1078
 - predictor, 316
- Maximum partial likelihood estimates (MPLEs), 923, 924
- Maximum-flow model, 647–648
- McDiarmid's inequality, 904
- Mean deviations, 48, 49
- Mean function, 69, 70, 77
- Mean life, 1014
- Mean lifetime, 1046
- Mean past life (MPL), 396
- Mean residual life (MRL), 11, 55, 396
- Mean squared prediction error (MSPE), 783
- Mean square error (MSE), 100, 101, 323, 596, 713, 714, 718, 868
- Mean time before failure (MTBF), 284, 289
- Mean time between failure (MTBF), 142–144, 968
- Mean time between software failures (MTBF), 447
- Mean time to failure (MTTF), 9–10, 55, 1010
- Mean time to repair (MTTR), 425
- Mean-value function, 146, 481, 484, 487–488
- τ -Measure of dependence, 850
- Measure of dependence, 849–851
- Measurement errors (ME), 70, 77, 82, 83, 335, 605, 607
- model, 345
- Measurement system analysis (MSA), 243, 245
- Mechanics, 1009
- analysis, 1004

- Median, 5, 48
- Median of absolute deviation (MAD), 613
- Memory organizations, 991
- Memoryless property, 13, 14
- Mesh, 1012
- M*-estimates, 613
 - asymptotic relative efficiency, 1121
 - distribution and calculation of, 1117
 - distribution of Procrustes model, 1119
 - hands data, 1118
 - Σ matrix and geometry, 1118
 - Procrustes model, 1117
 - statistical assumptions for Procrustes model, 1119
 - statistical assumptions of spherical regression, 1123–1124
 - statistical properties of, 1118–1121
- Meta-analysis, 580
 - of microarray, 632
- Method of least squares, 32
- Method of moments, 25
- Metric dimension, 907
- Metric entropy, 907
- Metric space, 906
- Metropolis algorithm, 777
- Metropolis ratio, 777
- Metropolis-Hastings algorithm, 773, 775, 777–779
- Metropolis-Hastings (MH) sampling, 968
- Metropolis-within-Gibbs, 116
- Microarray data, 622
 - four-level analysis of, 622
 - fourth-level analysis, 632
 - second-level analysis of, 623–625
 - third-level analysis, 625–632
- Mills seeding model, 433
- Minimal cut, 8
- Minimal path, 7
- Minimal repair, 187–189, 544, 546
- Minimization approach, 737–739
- Minimizer, local, 679
- Minimum cut sets, 62
- Minimum density power divergence estimators (MDPDEs), 1050–1052
- Minimum mean squared error, 202
- Minimum path sets, 62
- Minimum spanning tree model, 643–644
- Minimum variance unbiased estimator, 23
- Minimum-cost-flow model, 648
- Minorants, convex, 682
- Minorization, affine, 680
- Misclassification, 628
- Missing data, 599
- Missing information principle, 705, 711–712
- Mixed EWMA-CUSUM chart, 271
- Mixed logit model, 1081–1084
- Mixed sampling plans, 233
- Mixed-level foldover designs (MLFODs), 306, 308
- Mixed-type and multi-modality data, 216
- τ -Mixing condition, 852, 853
- Mixing conditions, 849, 858
- Mixture representation, 938
- MO-NH distribution, 46
- Mode, 5
- Mode-*m* matrixization, 818, 823
- Mode-*m* matrix product, 818
- Model comparison, 782–787
- Model integration, 952–955
- Model mis-specification analysis, 1046–1047
- Model robustness and sensitivity analysis, 781
- Modifications of the desirability function, 212
- Modifications of traditional methods, 201
- Modified chain sampling plan (MChSP-1), 225–226
- Modified Cholesky decomposition (MCD), 888, 897
 - adaptive banding, 890–892
 - for banded matrix estimation, 890
 - for covariance matrix estimation, 889–890
 - for linear discriminant analysis, 895
 - ordering issue, 892–893
 - for portfolio optimization, 894–895
 - for precision matrix estimation, 888–889
 - varying covariance matrix estimation, in multivariate time series, 893–894
- Modified ensemble transform Kalman filter (ETKF) modification method, 954
- Modified replacement, 559–562
- Modified Sarhan-Balakrishnan bivariate distribution, 754–755
- Molten lava, 1066
- Moment(s), 47–48
 - generating function, 49
 - inequalities, 5–6
- Moment-based estimators, 824, 826
- Monitoring batch means, 203
- Monitoring convergence, 780–782
- Monitoring information, 539
- Monitoring magnitude, 271
- Monitoring methods, 279
- Monotone likelihood
 - Cox proportional hazards regression model, 924
 - piecewise constant hazard regression model, 924
- Monotone partial likelihood, 923
- Monotonic decreasing bases (I-splines), 113
- Monotonic function, 903
- Monotonicity, 1097
- Monte Carlo approximation, 774
- Monte Carlo-based method, 968
- Monte Carlo (MC) estimators, 785, 972–974, 977, 978
- Monte Carlo sample size, 774
- Monte Carlo sampling, 585, 773–774
- Monte Carlo simulation, 110, 115, 323
- Monte Carlo standard error (MCSE), 774
- Monte-Hall problem, 765
- Mood test, 372
- mRNA, 628
- m*-th order auto-covariance matrix, 862
- Multi-agent financial decision support system, 996–997
- Multi-agent system, 983
- Multi-component, 527
- Multi-criteria decisions, 981
- Multi-level, 83
- Multi-level trend renewal process (MTRP), 106, 115–117, 125
- Multi-source data, 950
- Multi-stage logistics model, 664–665
- Multicollinearity, 595
- Multidimensional mixed sampling plans (MDMSP), 233
- Multidimensional multivariate distribution, 955–957
- Multidimensional OLAP (MOLAP), 800
- Multilevel/hierarchical functional data, 68
- Multinomial distribution, 766
- Multinomial logit model (MNL), 1078, 1083
- Multiobjective genetic algorithm (MoGA), 639
- Multiobjective optimization problems, 639–641
- Multiobjective transportation, 663
- Multiple additive regression tree (MART), 803
- Multiple components, 1052

- Multiple linear regression, 89, 99
 - Multiple outliers, 606
 - Multiprocessor scheduling model, 656–657
 - Multisource data, 950–952
 - Multistage process planning (MPP) model, 649–650
 - Multistage variation propagation, 812
 - Multivariate adaptive regression splines (MARS), 803
 - Multivariate analysis, 616
 - Multivariate ASN (MASN) distribution, 94
 - Multivariate CDs, 580
 - Multivariate characteristics, 205–206
 - Multivariate chart using real-time contrast, 206
 - Multivariate cumulative sum (MCUSUM), 940
 - Multivariate CUSUM chart, 206
 - Multivariate data, 932
 - Multivariate defect modeling, 812
 - Multivariate EWMA chart, 206
 - Multivariate exponentially weighted moving average (MEWMA), 940
 - Multivariate FPCA, 84
 - Multivariate geometric skew normal (MGSN) distribution, 94–95
 - Multivariate normal, 88, 94, 95
 - distribution, 110, 1119, 1123, 1124
 - Multivariate process control, 940–942
 - Multivariate quadratic loss, 211
 - Multivariate response envelope model, 821–822
 - Multivariate response regression, 821
 - Multivariate T^2 chart, 206
 - Musa execution time model, 432
 - Musa-Okumoto logarithmic Poisson execution time model, 433
 - Mutually independent, 7
- N**
- Nagaev-type inequality, 854
 - for linear processes, 852
 - National vulnerability database (NVD), 478
 - Natural language processing, 984, 994
 - Naïve Bayes algorithm, 870
 - Near-zero-defect manufacturing, 266
 - Nearest neighbor (NN), 806
 - Negative binomial distribution, 264
 - Negative log-likelihood function, 822
 - Negatively skewed, 90
 - Nelder–Mead algorithm, 515, 515
 - Nelson model, 433
 - Nelson-Aalen estimator, 832, 836, 842, 844
 - Nelson-Aalen formula, 836
 - Nested logit model, 1081–1082
 - Nests, 1081
 - Network architecture, 805
 - Network reliability design, 661
 - Neural function, 630
 - Neuroimaging studies, 817
 - Newton-Raphson, 89, 97
 - algorithm, 1098
 - method, 709
 - procedure, 162
 - Ng-Kundu-Chan model, 155
 - No fault found (NFF) problem, 214
 - Node(s), 994
 - impurity, 631
 - Nonconvex analysis, 676
 - Non-destructive testing (NDT), 971
 - Nondominated Sorting Genetic Algorithm II (NSGA II), 640
 - Nonhomogeneous Poisson process (NHPP), 106, 148, 432, 459, 481, 544
 - models, 447
 - software reliability models, 482–483
 - Noninformative prior, 765, 766, 970
 - Non-linear optimization, 89
 - Non-normal multivariate distributions, 932
 - Nonoverlapping batch means, 203
 - Non-parametric empirical Bayes, 768
 - Nonparametric fiducial inference, 583–585
 - Non-parametric learning, 868
 - Nonparametric maximum likelihood estimator, 62, 1047
 - Non-parametric methods, 955
 - Nonparametric regression modeling, 802
 - Nonparametric regularization, 950
 - Nonparametric risk analysis, 956
 - Nonparametric smoothing, 69
 - Non-parametric SPC, 362, 369, 379
 - Nonparametric tolerance limits, 35
 - Nonrepairable degraded systems
 - numerical example, 502–506
 - three competing process, 500–502
 - two competing process, 497–499
 - Non-symbolic structure, 993
 - Non-zero correlations, 1081
 - Norenewable warranties, 186
 - Normal cone, 679
 - Normal copula, 935
 - Γ -normal distribution, 45
 - Normal distribution, 15–16, 90, 99, 490
 - Normal parameters, 33
 - Normalized, 73, 77
 - criteria distance, 492, 493
 - Norris-Landzberg (N-L) formula, 1006
 - Numerical integration, 71, 772
 - Numerical simulation, 1004
- O**
- Object detection, 813
 - Observations and measurements, 950, 951
 - Observed Fisher information matrix, 711, 729
 - Occupancy probability, 787
 - Odd log-logistic-G, 50
 - Odds, 1078
 - O-function, 50
 - One dimensional Lebesgue measure, 739
 - One-by-one inspection process, 265
 - One-dimensional joiner, 132
 - One-shot device, 1039
 - CSALTs and models, 1040–1043
 - likelihood inference, 1043–1050
 - with multiple failure modes, 1052–1054
 - SSALTs, 1054–1056
 - testing data, 1041
 - One-step Newton-Raphson method, 1044
 - Online analytical processing (OLAP), 800
 - Online stochastic DCA, 688–696
 - Open-source software (OSS), 443
 - development paradigm of, 445
 - reliability of, 444–446
 - stochastic differential equation modeling, 448
 - Operating characteristic (OC) curve, 222
 - Operational coverage, 463
 - Operational effort functions, 464

- Operational patch time release policies, 468
 - assumptions, 468
 - for faults and vulnerabilities, 473
 - software with warranty, 469
 - vulnerable software systems, 476–479
 - Opportunistic scheme, 517
 - Optimal classifier, 826
 - Optimal proportion, 1055
 - Optimal stopping, 1096
 - Optimal warranty policy, 192
 - Optimization, 386, 823, 1102
 - model description, 397
 - model dormulation, 397
 - of multivariate loss functions, 211
 - problem parameters of subsystems, 400
 - Optimum design, 106, 119
 - Oracle VM Virtual box, 466
 - Order statistics, 6
 - Ordering-averaged method (OAM), 892–893
 - Ordering of variables, 888, 892, 894, 897
 - Ordinary least squares (OLS) estimator, 822, 823
 - Ordinary logistic regression (OLR), 593–595
 - Ordinary moments, 48
 - Ordinary renewal process, 147
 - Organization of memory, 991
 - Orlicz ψ_2 norm, 862
 - Orthogonal matrix, 1060, 1112
 - Orthogonality, 78
 - Orthonormal, 71
 - Out-of-control, 362
 - Outlier(s), 605, 608–609
 - detection, 83
 - groups of outliers, 610–613
 - large data sets, 614–615
 - robust regression, 613–614
 - single outlier, 608–610
 - Output, 1092
 - Over-dispersion, 268
 - Overfitting, 875, 902
 - Overlapping batch means, 203
- P**
- Package life, 1017
 - Packaging machines, cigarettes production, 287–289
 - Pairwise dependent, 135
 - Palmgren–Miner rule, 970
 - Parallel configuration, 139, 140
 - Parallel repairable system, 141
 - Parallel–series systems, 188
 - Parallel structure, 419
 - Parallel system, 7, 61, 137, 316, 544
 - Parameter, 712
 - uncertainty, 1009
 - variation, 1005
 - Parameter estimation, 23–25
 - confidence intervals, 488–490
 - maximum likelihood estimation, 486–487
 - mean-value function, 487–488
 - software reliability, 488
 - Parametric and semi-parametric models, for survival data, 832–833
 - Parametric bootstrap confidence intervals, 1046
 - Parametric empirical Bayes, 768
 - Parametric function, 712
 - Parametric learning, 867
 - Pareto distribution, 20
 - Pareto solution, 639
 - Paris' equation, 972
 - Paris' model, 972
 - Parsimonious representation, 822
 - Partial independence, 130
 - Partial least squares (PLS), 824, 825
 - Partial likelihood function, 923
 - Particle swarm optimization (PSO), 636
 - Pascal distribution, 13
 - Patch release time, 459
 - Path vector, 7
 - PBGA 316, 1011
 - PC scores, 71, 72
 - Pearson correlation coefficient, 1015
 - Pearson's linear correlation coefficient, 935
 - Peck's model, 1031
 - Penalization, 929
 - Penalized partial likelihood function, 929
 - Penalty
 - exact, 682
 - function, 821
 - Percentile residual life, 55
 - Perceptron, 866
 - learning, 874
 - Perfect dependence, 934
 - Perfect repair, 187
 - Performance, 522
 - criteria, 492
 - degradation, 942
 - evaluation function, 998
 - measure independent of adjustment, 209
 - measures, 209
 - of models, 1085
 - Performance analysis
 - of Poisson and binomial models, 299
 - of statistical methods, 296
 - Periodic inspection planning, 525
 - Periodic replacement, 543, 544, 546–547, 552–560, 562, 564–565
 - Permutation test, 623
 - Personal choice model, 1085
 - Personal preferences, 1082
 - Perturbation plot, 71
 - Pham distribution, 20
 - PHM-geometric (PHMG) distribution, 756
 - Piecewise constant hazard regression model, 922, 924, 928
 - Pixel data-driven approach based on software GUI, 448–450
 - Plackett-Burman design, 306
 - Plastic ball grid array (PBGA) array, 1011
 - Point availability, 143
 - Point estimation, 23, 578
 - univariate case, 768
 - Poisson distribution, 12–13, 145, 262, 265, 771
 - Poisson model
 - stock level conditioned to minimal availability method, 284–285
 - stock level conditioned to minimum total cost, 285–286
 - Poisson process, 145, 269
 - Poisson regression, 771
 - Poka-Yoke (mistake-proofing), 244
 - Polar projection, 1072
 - Polish space, 850
 - Political risk, 988
 - Polyhedral DC program, 681
 - Polymer composites fatigue testing, 122–125
 - Population model, 825
 - Portfolio optimization, 887, 894–895
 - Positive definite, 888

- Positive FDR (pFDR), 624
- Positively skewed, 90
- Positivity condition, 778
- Posterior covariance, 770
- Posterior covariance matrix, 788
- Posterior distribution, 533, 765, 766, 768
- Posterior mean, 769
- Posterior mode, 769
- Posterior model probabilities, 786
- Posterior predictive checking, 781–782
- Posterior predictive density, 770, 781
- Posterior predictive loss, 784
- Posterior predictive risk, 784
- Posterior probability, 624
- Posterior propriety
 - Cox proportional hazards regression model, 925
 - piecewise constant hazard regression model, 925
 - remedies for, 925
- Potentially diverges with n , 854
- Power law function, 461
- Power model, 1030
- Power normal, 88
- Power supplies, 144
- p -quantile, 533
- Precision matrix, 887, 889, 897
 - estimation, 888–889
- Prediction, 73, 316, 317, 328
 - likelihood functions, 317
 - maintenance, 528
 - maintenance modeling, 540
 - problems, 163
 - residuals, 607, 610, 611, 614
- Prediction interval (PI), 108, 111
- Predictive-ratio risk, 492
- Predisposition effect, 987
- Preventive maintenance, 187, 496, 516, 531
- Preventive threshold, 526
- Principal Analysis by Conditional Estimation (PACE), 70, 72, 82
- Principal components, 70
 - algorithms, 952
 - analysis, 622, 877, 887
- Principal differential analysis, 84
- Printed circuit board (PCB), 242, 798
- Prior and posterior distributions
 - Cox proportional hazards regression model, 924
 - piecewise constant hazard regression model, 924
- Prior density, 772
- Prior distribution, 530, 765, 767–769, 771, 785, 789, 1100
- Priority-based genetic algorithm (priGA), 658
- Pro-rata warranty (PRW), 187
- Probabilistic inference, 966
- Probabilistic model-based clustering (PMC), 626, 628
- Probabilistic modeling, 966–967
 - probabilistic parameter estimation – inverse problem, 967–968
 - probabilistic prediction – forward problem, 968
- Probabilistic parameter estimation, 967–968
- Probabilistic rational model (PRM), 622
- Probability, 1092
 - density, 784
 - distribution, 966
 - limit, 262, 266, 271, 272
 - mass function, 158
 - plotting, 57
 - ratio, 1081
 - weighted moments, 47
- Probability and moment inequalities, dependence, 851–852
 - sample sum for random matrices with $d_n \geq 1$, 856–857
 - sample sum for random vectors with $d_n \geq 1$, 854–856
 - sample sum for scalars with $d_n = 1$, 852–854
 - U- and V-statistics, 857–858
- Probability density function (PDF), 4, 7, 15, 43–45, 51, 87, 88, 90–92, 94–96, 107, 110, 118, 484, 705, 933, 936, 967–970, 1041
 - for weibull distribution, 418
- Probability of detection (POD), 971
- Probability of failure (POF), 968, 970, 971, 973, 977
- Probit model, 1077
- Probit regression, 771
- Process capability analysis, 245
- Process capability indices (PCI), 243
- Process control, 787, 788, 932, 940–942
- Process distribution, 362
- Process of learning, 982
- Process prediction, 204
- Process/product capability index (Cp), 245
- Procrustes model, 1112
- Procrustes regression model
 - large sample asymptotics, 1064–1065
 - and spherical regression, 1060–1061
- Product failures, 187, 190
- Production process, 222
- Product reliability, 190
- Profile likelihood function, 924
- Profile monitoring, 207
- Prognostics and health management, 213–214
- Progressive censoring methods, 153
 - accelerated life testing, 171
 - ageing properties, 160
 - coherent systems to progressively type II censored order statistics, 164
 - competing risk modelling, 170
 - conditional density function, 166
 - counting process approach, 163
 - dispersive order, 160
 - experimental design, 164
 - inference, 161
 - joint probability mass function, 166
 - joint progressive Type-II censoring, 174
 - lifetime performance index, 171
 - likelihood inference, 162
 - log-concavity and unimodality properties, 158
 - MLE under progressive Type-II censoring, 167
 - multivariate stochastic order, 159
 - point estimation, 161
 - prediction problems, 163
 - progressive hybrid censoring, 168
 - progressive type-I censoring, 165, 167
 - progressive type-II censoring, 168
 - reliability based on progressively type-II censored data, 170
 - reliability sampling plans, 171
 - simple step-stress testing, 172
 - stage life testing model, 172
 - standard models of, 171
 - statistical intervals, 163
 - statistical tests, 164
 - stress-strength models, 170
 - stress-strength reliability, 170
 - type-I and type-II, 154
 - versions of, 154
- Progressive censoring signature (PC-signature), 165
- Progressive hybrid censoring, 155
- Progressive type-II censoring, 154

- Projection matrix, 825
 - Proportional hazard class, 735–736
 - Proportional hazards assumption, 1050
 - Proportional hazards models, 1043, 1050
 - Proportional-integral-derivative, 202
 - Proportional reversed hazard class, 736
 - Prostate-specific antigen (PSA), 923
 - Proximal point algorithm, 683
 - Proximal upper bounds, 683
 - PRRM, 1080
 - Pruning, 363, 370
 - data, 371
 - Pseudo log-likelihood, 97, 99
 - Pseudo-observations, 940
 - Pseudometric space, 907
 - Psychological behavior, 983
 - Public health surveillance, 813
 - Pugh concept, 249–250
 - p -value, 362, 577, 596
 - Python, 809
- Q**
- QR decomposition, 770
 - Quadrant reduced model, 1012
 - Quadratic discriminant analysis, 870
 - Quadratic upper bounds, 683
 - Quadric time-dependent cost function, 1097
 - Quadstone, 809
 - Quality engineering, 255
 - Quality function deployment (QFD), 243, 244, 249
 - Quantile, 54
 - estimator, 970
 - function, 57
 - values, 955
 - Quantile–quantile plot, 625
 - Quantitative methods, 279
 - Quantitative reliability, 1009
 - Quasi-MLE (QMLE), 1046, 1047
 - Quasi-Newton method, 685
 - Quasi-renewal process, 147–148, 188, 192
 - Quaternions, 1070–1071
 - Quick switching sampling (QSS) system, 229
 - Quick, unbiased and efficient statistical tree (QUEST), 809
- R**
- Rademacher complexity, 902, 904–905, 909–911, 916
 - Radial basis function (RBF), 806
 - Radon measure, 687
 - Ramamoorthy and Bastani model, 433
 - Random coefficient(s), 60, 1084
 - degradation path, 497
 - Random effects, 110
 - Random environment, 491
 - Random field environment, 483–484, 486, 488
 - Random forest, 631, 871
 - theoretical properties, 844–845
 - Random forests, for right-censored survival data
 - ensemble of trees with randomness, 836–837
 - ensemble prediction, 837
 - evaluation criteria for survival models, 838–839
 - out-of-bag error, 839
 - tuning parameters, 837–838
 - Randomized logistic degradation path, 497
 - Random number generator, 789
 - Random oscillation, 112
 - Random regret minimization (RRM), 1076, 1077, 1079–1080, 1084
 - Random replacement, 545
 - Random utility maximization (RUM), 1075–1079
 - Random variable, 49, 1099
 - Random walk Metropolis algorithm, 777
 - Random warranty cost, 193
 - Rare event, 973
 - Rate of return, 998
 - Ratio of maximized likelihood (RRML) approach, 316, 322
 - Rayleigh distribution, 20
 - Rayleigh probability density, 20
 - Reactive agents, 992
 - Read-write errors, 268
 - Real data analysis
 - LS-SGSN model, 101–102
 - SGSN regression model, 102
 - Reciprocal importance sampling estimator, 786
 - Recognition and reward systems, 242
 - Recurrence formula, 131
 - Recurrence transition, 130
 - Recurrent event data analysis, 115
 - background and data, 115–116
 - MTRP model and parameter estimation, 116
 - prediction for component events, 117
 - Recurrent neural network (RNN), 874
 - Recursive function, 1092
 - Recursive partitioning, 603
 - Recursive reliability function, 1090
 - Redundant replacement, 544, 562
 - policies, 562–565
 - Reference alternative, 1078
 - Reference prior, 766–767
 - References on dynamic robust design, 212
 - Regeneration times, 535
 - Regenerative process, 535
 - Regionalization of climate change models, 952, 953
 - Regression, 767
 - analysis, 1017
 - coefficients, 888, 889
 - function, 78
 - Regression model, 824, 825
 - residuals and leverage in, 607–608
 - Regret, 1079
 - functions, 1084
 - Regularization, 875
 - Regularized estimator, 820, 821
 - Regularized incomplete beta function, 390
 - Reinforcement learning, 867
 - Rejection probability, 777
 - Rejection sampling, 774
 - Relational database system, 799
 - Relational OLAP, 800
 - Relative advantage, 1081
 - Relative advantage maximization (RAM), 1076, 1077, 1080–1081
 - Relative bias, 1047
 - Relative efficiency, 713
 - Relative error (RE), 974
 - Relative humidity (RH), 112
 - Relative precision, 122
 - Reliability, 9, 53, 141, 142, 213, 544, 789, 966, 968–969, 1046, 1091, 1092
 - analysis, 388, 922
 - analysis methods, 443
 - approximation, 1093–1094
 - availability, 425
 - characteristics, 393–397

- Reliability (*cont.*)
- characteristics of software, 445
 - component, 404
 - damage tolerance model, 971–974
 - decline in, 404
 - definitions of, 445
 - engineering, 1010, 1015
 - function, 129, 498, 586
 - improvement, 392
 - maintainability, 425
 - modeling, 190
 - optimization, 660–661
 - performance, 188
 - prediction, 108, 110–111, 114, 489–490
 - quality management, 443
 - repairable complex systems, 404
 - repairable system, 424
 - safe-life model, 969–971
 - software, 446
 - studies, 44
 - system, 404
 - theory, 789
 - time dependent functions, 1094
- Reliability design problem, 659–662
- network reliability design, 661
 - redundant unit and alternatives, 661
 - reliability optimization, genetic algorithm for, 660–661
 - tree-based network design, 661–662
- Remaining life, 107, 111
- Remaining useful life (RUL), 523, 532–533
- distribution, 533
- Renewable warranty, 186
- Renewal, 536
- density, 148
 - distribution, 116
 - function, 146
 - process, 106, 146–147
 - theory, 535
- Rényi entropy of order, 396
- Repairable degraded systems
- inspection-based maintenance, three competing process, 511–516
 - inspection maintenance policy, 505–506
 - maintenance cost modeling, 506–509
 - numerical example, 509–511
 - optimization of the maintenance cost rate policy, 509
- Repairable products, 187, 190
- Repairable system, 404
- importance and sensitivity analysis on, 405–411
 - reliability, 424
- Repair-cost-limit warranty (RCLW), 188
- Repair-number-limit warranty (RNLW), 188
- Repair-time-limit warranty (RTLW), 188
- Repetitive group sampling (RGS) plan, 227
- Replacement first, 544, 547–549, 552–558, 560, 561, 563–564
- Replacement last, 544, 548–549, 552–555, 557–561, 563–565
- Replacement middle, 544, 561
- Replacement overtime, 544–547, 549–552, 554–557
- first, 549–550, 554–556
 - last, 550–552, 555–557
- Representation of the knowledge, 990
- Re-sampling, 1045
- Rescorla exponential model, 458, 468
- Residual, 596
- covariance matrix, 607
 - deviance, 594
 - sum of squares, 100, 594
- Resource-constrained project scheduling model, 656
- Resource-constrained project scheduling problem (rcPSP), 650
- Response surface method (RSM), 243, 250–251
- Response surface models (RSM), 210
- Response variable, 769
- Responsibility, 439
- Revealed preference, 1083
- Reversed hazard function, 736
- Reversed hazard rate, 55
- Reverse logistics model, 666–667
- Reversible, 775, 777
- Reversible jump Markov chain Monte Carlo (RJMCMC), 779
- Ridge regression, 875
- r-independence, 135
- Right censored, 57
- data and survival analysis, 831–833
- Right-censoring, 1040
- Risk of interchangeability, 989
- Risk of redemption on demand, 989
- Risk of reinvesting, 989
- Risk set, 111
- Risk-neutral manufacturer, 191
- Robust adversarial learning, 909–911
- Robust design, 200
- applications of, 213
 - dynamic, 212–213
 - multiple responses, 211–212
 - single responses, 207–211
- Robust estimation, 606, 610, 613
- Robust regression, 606, 613–614
- Robust reliability design, 64
- Robustness, 912
- Rolling-horizon, 527
- Root mean square error (RMSE), 450
- Root- n consistent, 82
- R package copula, 944
- μ RRM, 1080
- R-square, 78
- RTREE, 631
- S**
- Saddle point approximation technique, 1093
- Safe-life, 965, 969–971
- Salford Systems, 809
- Sample auto-covariance matrices, 855
- Sample moments, 25
- Sample validation, 783
- Sarhan-Balakrishnan bivariate distribution, 753–754
- Satellite model, 959
- Sauer-Shelah lemma, 906
- Scalar-on-function regression, 82
- Scale density, 766
- Scheduling problem, 652–659
- Schick and Wolverson (SW) model, 430
- Score statistic, 783
- Screening-optimizing continuum, 305
- Search algorithm, 892
- Seasonal pattern, 112, 114
- Seasonal regression model, 283
- Second-order D-efficiency, 308
- Selection theorem, 688
- Self-adaptive, 526
- Self-organization maps (SOM), 626, 809
- Self-validation, 439
- Semantic network, 997

- Semantic network processing system (SNePS), 994
- Semicontinuity property, 691
- Semimartingale convergence theorem, 695
- Semi-regenerative, 536
 - process, 535
- Semi-supervised learning, 867, 878
- Sensitive observations, 606
- Sensitivity, 1101
 - matrix, 605, 606, 612–614, 616, 617
 - Peña's statistics, 614
- Sensitivity analysis
 - extrinsic sensitivity, 1103
 - importance and, 404, 408
 - intrinsic sensitivity, 1102
- Separable covariance, 818, 823
- Sequential Bayesian design (SBD), 106, 117, 119, 120
- Sequential probability ratio test (SPRT), 263
- Sequential sampling, 36
- Sequential test planning
 - dual objectives, 121–125
 - of accelerated life tests, 117–120
- Serial correlation and effective sample size, 780
- Series system, 7, 61, 419
- Service level optimization, 289–290
- Set-covering model, 644
- Shallow analysis, 984
- Shannon entropy, 396
- Shape-restricted splines, 113
- Shared load parallel system, 141, 142
- Sharpe ratio, 998
- Shewhart chart, 262, 362–366, 379
- Shewhart control, 201
 - chart, 334
- Shewhart GZIP chart, 269
- Shifted Jeffreys-type prior, 928
- Shock model, 734
- Shopping location choices, 1083
- Short-range dependence, 852
- Short-term, 529
 - horizon, 527
- Shortest-path model, 646–647
- Siegmund's approximation, 338
- Signal compression, 84
- Signal processing, 817
- Significance analysis of microarrays (SAM), 624–625
- Significance tests, 594
- Significantly differentially expressed genes, 623
- Simple assembly line balancing (sALB), 658, 659
- Simple step-stress model, 705–706
- Simulation study, 713, 714, 718
- Simultaneous confidence band, 83
- Simultaneous shifts, 267
- Single nucleotide polymorphism, 622
- Single outlier, 608–610
- Single patch model, 475
- Singular distribution, 742
- Singular value decomposition, 819, 1114
- Six Sigma
 - Academy, 241
 - description, 240
 - Executive, 241
 - failures, 242
 - implementation, 241–242
 - training and belt structure, 241–242
- Skew normal, 87
- Skew-symmetric matrix, 1116
- Skewness, 594
 - coefficient, 5
- Skip-lot sampling plans, 230
- Sklar's theorem, 933
- Slice model, 1012, 1013
- Slice sampler, 779
- Slow moving demand, 276
- Smallest extreme value distribution, 418
- Smart meters, 879
- Soccer model, 734
- Social network choice protocol, 1085
- Socio-environmental security, 950, 951, 958
- Soft failure, 114
- Software buyers growth function, 461
- Software data mining (DM), 798
- Software failure, 445, 481
 - data, 492
- Software fault, 445
- Software fault-detection/failure-occurrence phenomena, 444
- Software life cycle models, 439
- Software reliability
 - correctness of, 428
 - definition, 426
 - growth models, 443
 - vs. hardware reliability, 427
 - measurement of, 428
 - taxonomy of, 428
- Software reliability models, 29–30, 429, 430, 481, 483, 488
 - calibration model, 482
 - NHPP, 482–483
 - RFE models, 484–486
 - time between failure models, 430
- Software requirements analysis, 438
- Software testing, 146, 428, 435, 481, 483, 486
 - definition, 435
 - optimal time for, 437
 - process, 436
 - psychology of, 436
- Software validation, 438
 - after change, 439
 - benefits, 440
 - as part of system design, 438
 - plan, 439
 - principles of, 438
 - process, 439
- Software verification and validation, 437
- Sparse design, 69
- Sparse functional data, 69
- Sparsity, 888–891, 896, 897
- Spatial visual system (SVS), 993
- Spearman correlation, 626
- Spearman's rho, 934
- Special cause, 361
 - charts, 202
- Special orthogonal matrix, 1112
- Spherical regression
 - differential manifold, 1062
 - large κ asymptotics, 1065–1066
 - least squares estimation, 1061
 - model, 1112
 - and Procrustes model, 1060–1061
 - tangent space approximation, 1063
- Spline-based approaches, 69, 70
- Standard accuracy, 912
- Standard age replacement, 545
- Standard deviation (SD), 5, 98, 102, 894

- Standard Dudley's entropy integral, 915, 917
 Standard error (SE), 729, 895–897, 923, 925
 Standardized influence functions (SIF), 1125
 Standardized sensitivity vector, 614
 Standardized time series, 203
 Standard normal, 88, 90, 98
 distribution, 15, 16
 State, operator and result (SOAR), 993
 Stationary, 775, 777
 distribution, 780
 law, 536, 537
 measure, 535
 probability distribution, 536
 process, 114
 Stationary Gaussian vector autoregressive model, 859
 Statistical applications, 861–862
 Statistical based reliability test (SRT), 1025
 Statistical error analysis, 1069
 Statistical inference, 939–940
 Statistical learning, 901–902
 deep learning, learning theory for, 907–909
 logistic regression, generalization bounds for, 911–912
 robust adversarial learning, learning theory for, 909–911
 uniform convergence, 903–907
 Statistical Package of Social Sciences (SPSS) tool, 465, 478, 809
 Statistical process control, 200, 232, 244, 246, 256, 257, 266, 271, 333, 361, 810
 for autocorrelated processes, 201–203
 automatically controlled processes, 204
 vs. automatic process control, 203
 charts, 204
 efficiency vs. robustness, 205
 GLRT-based multivariate methods, 202
 for i.i.d. processes, 200–201
 for multivariate characteristics, 205–206
 for profile monitoring, 207
 Statistical property, 1019
 Statistical quality control, 788
 Steady state availability, 143, 144
 Step-stress ALTs (SSALTs), 1040, 1054–1056
 Step-stress model, 704–706
 Stirling polynomials, 47
 Stochastic approximation, 687
 Stochastic composite function, 686
 Stochastic DC programming and DCA, 687–696
 Stochastic dependence, 62–63
 structure, 130
 Stochastic expectation-maximization (St-EM) algorithm, 704, 705, 709–710
 Stochastic model-based algorithms, 686
 Stochastic optimization, 685–687
 Stochastic processes, 69, 137, 145, 522–525
 Stochastic simulation, 774
 Stochastic subgradient method, 685
 Stock exchange market risk, 988
 Strain, 1012
 Strehler-Mildvan model, 516
 Strength Pareto Evolutionary Algorithm 2 (SPEA2), 640
 Stress, 117, 1012
 levels, 124
 model, 734
 Stress intensity factor (SIF), 972
 Stress-life, 969
 Stress-strength models, 170
 Strong convexity, 677
 Strong criticality, 679
 Strong law of large numbers, 776
 Structure function, 7
 Structured query language (SQL), 799
 Studentized residual, 607
 Student's t distribution, 17
 Student's T-test, 622
 Subanalytic data, 681–682
 Subgradient, 677
 Subjective prior information, 789
 Subsystem events, 115
 Successive convex approximation (SCA), 682, 683
 Successive DC decomposition, 696
 Sufficient dimension reduction, 821
 Supervised learning, 801, 867, 901
 Supply chain, 64
 Supply chain management (SCM), 664–666
 Supply chain network (SCN), 664
 Support function, 679
 Support vector machines (SVM), 629, 805–806, 869, 902
 Support vector regression, 876
 Supremum log-rank test, 834–835, 844
 Surface uncertainty
 probabilistic LCF life prediction, 975–977
 stress and strain states of Curvic couplings, 974–976
 Survival analysis, 44, 50, 787, 836–838, 842
 and right-censored data, 831–833
 Survival copula, 944
 Survival estimation, 832
 Survival forests
 bias and corrections of, 842–844
 Survival function (SF), 705, 706, 832, 833, 836–838, 841, 842, 844, 845, 944
 Survival signature
 interval of, 404
 limitations of, 403
 methodology, 404
 structure function and, 404
 Swamping, 606
 Symbolic architectures, 990–991
 Symbolic structure, 993
 Symmetric GSN (SGSN) regression model, 98–100
 real data analysis, 102
 simulation results, 101
 System maintenance, 512
 System mean time to failure, 144
 System regret, 1080
 System reliability, 61–64, 144, 385–387, 390, 419
 System state space, 501
 Systems health monitoring and management (SHMM), 214–216
- ## T
- Taguchi, G., 811
 Taguchi robust design methods, 811
 Taguchi's dynamic robust design, 212
 Taguchi's product arrays, 207–208
 Taguchi's signal-to-noise ratios, 209
 Tail dependence, 934
 Tangent space approximation, 1061–1064
 Taylor linearization, 1063, 1064, 1068
 Taylor series prediction method, 320
 T-chart, 269
 T^2 control chart, 941
 t-distributed Stochastic neighbor embedding (t-SNE), 878

- Technical analysis, 986
 - agents, 996
 - indicators, 987
 - Technological methods, 279
 - Tectonic plate reconstruction, 1066–1067
 - Hellinger criterion, 1067
 - statistical analysis of Hellinger estimate, 1068
 - Telecommunication(s), 811
 - software, 486
 - Temperature cycling test, 1007
 - Temperature-humidity model, 1031
 - Temperature ramps, 1014
 - Tensor classification, 825
 - Tensor data, 823
 - analysis, 818
 - Tensor discriminant analysis (TDA), 826
 - Tensor envelope partial least squares regression, 824–825
 - Tensor generalized linear model, 820–821
 - Tensor normal, 826
 - distribution, 819–820, 826
 - Tensor PLS algorithm, 824
 - Tensor predictor, 825
 - Tensor response linear model, 822
 - Tensor response regression, 820, 823, 824
 - Test
 - Case, 435
 - definition, 435
 - of independence, 597
 - planning, 106, 1050
 - Testing environment, 484
 - Testing life cycle, 436
 - Testing of hypotheses, 98
 - Text analysis agents, 996, 998
 - Text exploration, 984
 - The Houses of Quality (HOQ), 249
 - Theory of inventive problem solving (TRIZ), 249
 - Three competing processes
 - inspection-based maintenance fo repairable degraded systems, 511–516
 - inspection–maintenance policy, 511, 512
 - maintenance cost analysis, 512–513
 - nonrepairable degraded systems, 500–502
 - numerical examples, 514–516
 - optimization of cost rate, 514
 - Three Sigma program, 240
 - Three-Sigma limits, 262, 270
 - Threshold, 109
 - voting system, 1091
 - Tightened-normal-tightened sampling, 222
 - Time and effort, 439
 - Time-between-events, 262, 269–271
 - Time censoring, 1027
 - Time horizon, 279
 - Time series analysis, 861
 - Time to event data analysis
 - background and data, 109
 - model for covariates, 110
 - model for time to event data and parameter estimation, 109
 - reliability prediction, 110–111
 - Tire reliability study, 925–927
 - Tolerance design, 213, 305
 - Top-down learning, 991
 - Total probability, 4
 - Total quality management (TQM), 240, 242
 - Total time on test (TTT), 713, 718
 - Traditional reliability analysis
 - background and data, 107
 - reliability prediction, 108
 - time to event models and parameter estimation, 107–108
 - Transient analysis, 974, 975
 - Transition kernel, 775
 - Transition probability, 139
 - Traveling-salesman model, 645–646
 - Tread and belt separation (TBS), 925, 926
 - Tree-based models, 833–834
 - splitting mechanism, 834–836
 - tree prediction, 836
 - Tree-based network design, 661–662
 - Tree coefficient, 803
 - Tree estimator, 837
 - Tree prediction, 836
 - Trees and forests, 622
 - Trend-renewal process (TRP), 106
 - Tri-variate distribution, 131
 - Triangular arrays, 695
 - Triangular-array setting, 851
 - Triangular distribution, 22
 - Trivariate reduction method, 812
 - Trivariate replacement policy, 544
 - Tucker decomposition, 818
 - Two-attribute warranty, 189
 - Two competing processes, nonrepairable degraded systems
 - model description, 497
 - numerical example, 499
 - reliability evaluation, 498
 - Two-degradation-process, 500
 - Two-dimensional Lebesgue measure, 739
 - Two-dimensional warranty, 191
 - Two-level repairable system, 115
 - Two-parameter exponential distribution, 585–586
 - Two patch model, 476
 - Two-stage transportation problem (tsTP), 664
 - T-X family, 49–50
 - Type I censoring, 25
 - Type I error, 623
 - Type-II censoring, 25, 704, 706
- U**
- Ultraviolet (UV) radiation, 112
 - Unbiased estimator, 23
 - Uncertainty, 531, 675, 966
 - Under-dispersion, 268
 - Uniform convergence, 82, 902, 903
 - concentration inequalities, 903–904
 - covering number, 906–907
 - growth function and VC dimension, 905–906
 - Rademacher complexity, 904–905
 - Uniform distribution, 15
 - Uniform equicontinuity, 692
 - Uniform prior, 765
 - Unique vulnerabilities, 460
 - Universal representation of bivariate, 130
 - Unreplicated experiment analysis, 210
 - Unstructured data, 982
 - Unsupervised learning, 806–809, 867
 - clustering analysis, 876
 - DBSCAN, 877
 - dimension reduction techniques, 877–878
 - hierarchical clustering, 877
 - K-means clustering, 877

- Unsupervised learning (*cont.*)
 - principal components analysis, 877
 - semi-supervised learning, 878
- Upper and lower CDs, 577–578
- Upper-bound truncated exponential distribution, 15
- Upper control limit (UCL), 251
- Use-rate data, 109
- Used as good as new, 14
- Useful life period, 417
- User-dependent vulnerability discovery model, 460–461
- U-statistics, 845, 857
- Utility, 1076
- Utilization index, 288

- V**
- Validation, 425
 - coverage, 439
 - of software, 438
- Value at risk (VaR), 193
- Variable importance, 837, 838, 840–843
- Variable sample size (VSSI) CV control chart, 335
- Variable selection, 820, 839–841
- Variable-selection-based multivariate chart, 206
- Variance–covariance matrix, 488
- Variance matrix, 27
- Variance reduction method, 687
- Variation, 240
- Variational auto-encoders (VAEs), 912
- Variational Bayesian methods (VBM), 790
- Vector components, 952
- Vehicle routing procedures, 63
- Verification, 425, 1010
 - of software, 438
- Verification and validation activities (V&V), 425, 437
- Vibration acceleration models, 1031
- Virkler's fatigue testing data, 973
- Visual features, 805
- Vitali convergence theorem, 693
- Voice of customer (VOC), 248
- Voltage acceleration models, 1030
- Volterra equation, 537
- Voters, 1090
- Voting rule, 1090
- V-statistics, 857, 858
- Vtub-shaped failure rate, 22
- Vtub-shaped fault-detection rate, 482
- Vtub-shaped function, 492
- Vulnerability discovery models
 - effort-dependent, 461–462
 - generalized coverage-dependent model, 463–464
 - non-homogenous Poisson process, 459
 - notations, 458
 - user-dependent, 460

- W**
- Wald-type test(s), 1051–1052
 - statistics, 1051
- Warm standby system, 386
- Warranty, 185
 - claim, 188, 190
 - cost, 188, 189, 191
 - cycle, 186, 190
 - length, 192
 - management, 192
 - period, 188, 190, 192
 - service, 190
- Warranty policies
 - classification, 186–189
 - evaluation of, 189–193
- Warranty-maintenance, 188
- Warranty-management, 186
- Watanabe-Akaike information criterion (WAIC), 929
 - See also* Widely applicable information criterion (WAIC)
- Γ -W distribution, 51, 52
- Weak dependence case, 853
- Weak dependence measures, 850, 851
- Wearout period, 417
- Γ -Weibull density function, 45
- Weibull distribution, 18, 45, 46, 55, 57, 107, 110, 117, 270, 292, 418, 704, 706, 728, 737, 1028, 1041, 1047–1049, 1052, 1055
 - modified model extension, 292–293
 - simulation, 293–294
- Weibull models, 424, 1041–1042, 1055
- Weibull probability density, 18
- Weight(s), 1090
- Weight mapping crossover (WMX), 650, 651
- Weighted batch mean (WBM) method, 203
- Weighted Marshall-Olkin bivariate exponential distribution, 742
- Weighted threshold indecisive voting system
 - indecisive decision rule, 1099
 - indecisive model formulation, 1099
 - model assumptions, 1099
 - numerical example, 1100–1102
 - sensitivity analysis, 1102–1104
- Weighted threshold voting system
 - decision rule, 1091
 - in detection, communication and recognition, 1106
 - examples of system reliability, 1092–1093
 - future research, 1107
 - model assumptions, 1091
 - model formulation, 1091–1092
 - reliability analysis, 1093
- White-box testing, 437
- White type statistic, 1048
- Whöler curve, 56
- Widely applicable information criterion (WAIC), 782
- Wiener process, 524
- Wilcoxon rank sum test, 622
- Wilcoxon rank test, 577
- Wireless sensor network, 1107
- Working memory, 993

- X**
- XGBoost, 873
- XLMiner, 809

- Y**
- Yield modeling, 812
- Young inequality, 692
- Yule-Walker formula, 861

- Z**
- Zellner's *g*-prior, 925
- Zero-defect process, 262, 266, 271
- Zero-inflated models, 269
- Zero-inflated Poisson (ZIP) distribution, 266
 - CUSUM chart for, 267
 - EWMA chart for, 267–268
 - Shewhart chart for, 266–267
- Zero-inflation Poisson model, 266
- Zero-truncated Poisson distribution, 267